aws

Panduan Developer

Amazon Personalize

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Personalize: Panduan Developer

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Apa itu Amazon Personalize? 1
Harga untuk Amazon Personalisasi 2
Panduan untuk pengguna pertama kali 3
Menemukan Amazon Personalisasi dengan Mesin Film Ajaib 3
Menavigasi materi memulai dalam panduan ini 3
AWS Layanan dan solusi terkait 4
Layanan pihak ketiga 4
Pelajari selengkapnya 5
Bekerja dengan AWS SDKs 5
Amazon Personalisasi dan AI generatif 7
Rekomendasi dengan tema dari Content Generator 7
Metadata rekomendasi 8
LangChain Kode pra-konfigurasi untuk personalisasi 9
Cara kerjanya 10
Amazon Personalisasi alur kerja 10
Amazon Personalisasi istilah 12
Impor dan manajemen data 12
Pelatihan 15
Penyebaran model dan rekomendasi 18
Amazon Personalisasi data 19
Data interaksi 20
Data barang 20
Data pengguna 20
Data tindakan 21
Data interaksi tindakan 21
Menyiapkan Amazon Personalisasi 22
Mendaftar untuk Akun AWS 22
Buat pengguna dengan akses administratif 23
Wilayah dan titik akhir 24
Menyiapkan izin 24
Memberikan izin Amazon Personalize untuk mengakses sumber daya Anda 25
Memberikan izin kepada pengguna untuk mengakses Amazon Personalize 29
Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3 31
Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS 37

Menyiapkan AWS CLI 40
Menyiapkan AWS SDKs 41
Memulai tutorial 42
Memulai prasyarat 43
Membuat data pelatihan (Grup dataset domain) 43
Membuat data pelatihan (grup dataset kustom) 44
Memulai grup dataset Domain 45
Memulai grup dataset Domain (konsol) 46
Memulai grup dataset Domain (SDK for Java 2.x) 56
Memulai grup dataset Domain (SDK for Python (Boto3)) 65
Memulai grup dataset Domain (SDK untuk JavaScript v3) 71
Memulai grup Dataset Kustom 79
Memulai (konsol) 80
Memulai (AWS CLI) 92
Memulai (SDK for Python (Boto3)) 102
Memulai (SDK for Java 2.x) 108
Mencocokkan kasus penggunaan Anda dengan Amazon Personalize resource 121
Gunakan fitur kasing dan resep 122
Personalisasi waktu nyata 122
Eksplorasi 123
Pembaruan otomatis 124
Memilih kasus penggunaan 126
Kasus penggunaan VIDEO_ON_DEMAND 127
Kasus penggunaan ECOMMERCE 131
Memilih resep 135
Amazon Personalisasi jenis resep berdasarkan kasus penggunaan 136
Amazon Personalisasi resep 138
Melihat resep Amazon Personalize yang tersedia 140
Personalisasi pengguna-v2 140
Personalisasi Pengguna 144
Trend-Sekarang 160
Hitungan Popularitas 164
Personalisasi-peringkat-v2 164
Peringkat yang Dipersonalisasi 168
Barang Serupa 174
SIMS 177

Tindakan Terbaik Berikutnya- 184
Barang-Afinitas 189
Item-Atribusi-Afinitas 191
Resep warisan 193
Mempersiapkan data pelatihan 214
Pedoman format data massal untuk semua jenis data 215
Data interaksi item 216
Persyaratan data interaksi item 217
Data stempel waktu 218
Jenis peristiwa dan data nilai acara 219
Metadata kontekstual 221
Data tayangan 222
Contoh data interaksi 225
Metadata barang 226
Persyaratan data item 228
Data stempel waktu pembuatan 230
Metadata kategoris 230
Metadata teks tidak terstruktur 231
Data numerik 232
Data string non-kategoris 233
Contoh metadata item 233
Metadata pengguna 235
Persyaratan data pengguna 235
Metadata kategoris 236
Data string non-kategoris 236
Contoh metadata pengguna 237
Metadata aksi 238
Persyaratan data tindakan 239
Data stempel waktu kedaluwarsa tindakan 239
Ulangi data frekuensi 240
Data nilai 240
Data stempel waktu pembuatan 240
Metadata kategoris 241
Data string non-kategoris 241
Contoh metadata tindakan 241
Data interaksi aksi 243

Persyaratan data interaksi tindakan 244
Data tipe acara 244
Contoh data interaksi aksi 245
Membuat file JSON skema 247
Persyaratan pemformatan skema 247
Jenis data skema 248
Kumpulan data dan skema VIDEO_ON_DEMAND 249
Dataset domain VIDEO_ON_DEMAND dan persyaratan skema 250
Persyaratan kumpulan data interaksi item 251
Persyaratan dataset pengguna 253
Persyaratan dataset item 254
Kumpulan data dan skema ECOMMERCE 257
Dataset domain ECOMMERCE dan persyaratan skema 258
Persyaratan kumpulan data interaksi item 259
Persyaratan dataset pengguna 261
Persyaratan dataset item 262
Kumpulan data dan skema khusus 265
Dataset kustom dan persyaratan skema 266
Persyaratan skema kumpulan data interaksi item 269
Persyaratan skema kumpulan data pengguna 271
Persyaratan skema kumpulan data item 273
Persyaratan skema kumpulan data tindakan 275
Persyaratan skema kumpulan data interaksi tindakan 277
Daftar periksa kesiapan 280
Sudahkah Anda mencocokkan kasus penggunaan Anda dengan sumber daya Amazon Personalize? 280
Apakah Anda memiliki data interaksi item yang cukup? 281
Apakah Anda memiliki arsitektur streaming acara real-time? 281
Apakah data Anda dioptimalkan untuk Amazon Personalize? 282
Apakah Anda mengumpulkan data opsional yang dapat meningkatkan rekomendasi? 282
Apakah Anda memiliki rencana untuk menguji rekomendasi Anda? 283
Apakah Anda memiliki tujuan bisnis tambahan? 283
Membuat grup dataset 284
Membuat grup dataset (konsol) 285
Membuat grup dataset ()AWS CLI 286
Membuat grup dataset ()AWS SDKs 287

Membuat skema dan dataset 290
Membuat kumpulan data dan skema (konsol) 290
Membuat dataset dan skema ()AWS CLI 291
Membuat dataset dan skema ()AWS SDKs 293
Mengimpor data pelatihan 298
Mengimpor data massal 299
Mode impor 300
Membuat pekerjaan impor dataset (konsol) 301
Membuat pekerjaan impor dataset ()AWS CLI 303
Membuat pekerjaan impor dataset ()AWS SDKs 305
Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler 309
Informasi tambahan 311
Menyiapkan izin 311
Meluncurkan Data Wrangler dari Amazon Personalize 313
Mengimpor data ke Data Wrangler 314
Mengubah data 315
Menghasilkan visualisasi dan wawasan data 317
Memproses data dan mengimpornya ke Amazon Personalize 322
Mengimpor catatan individu 325
Mengimpor interaksi secara individual 326
Mengimpor pengguna secara individual 328
Mengimpor barang secara individual 333
Mengimpor tindakan secara individual 338
Menganalisis data pelatihan 342
Izin yang diperlukan untuk menganalisis data 343
Wawasan data 343
Melihat wawasan dan statistik kumpulan data 346
Rekomendasi domain 348
Status rekomendasi 349
Permintaan rekomendasi minimum per detik dan auto-scaling 350
Membuat rekomendasi domain 350
Membuat rekomendasi (konsol) 351
Membuat pemberi rekomendasi ()AWS CLI 353
Membuat pemberi rekomendasi ()AWS SDKs 354
Mengaktifkan metadata dalam rekomendasi 357

Mengkonfigurasi kolom yang digunakan saat pelatihan 358
Mengkonfigurasi eksplorasi untuk pemberi rekomendasi domain 361
Mengevaluasi rekomendasi domain 364
Mengambil metrik 365
Definisi metrik 368
Contoh 370
Sumber daya tambahan 370
Memperbarui pemberi rekomendasi 371
Memperbarui pemberi rekomendasi (Amazon Personalize console) 371
Memperbarui pemberi rekomendasi ()AWS CLI 372
Memperbarui pemberi rekomendasi ()AWS SDKs 373
Menghentikan pemberi rekomendasi 374
Menghentikan pemberi rekomendasi (konsol) 375
Menghentikan pemberi rekomendasi ()AWS CLI 376
Menghentikan pemberi rekomendasi ()AWS SDKs 376
Sumber daya khusus 379
Mengkonfigurasi solusi 380
Menciptakan solusi 381
Mengkonfigurasi pelatihan otomatis 389
Mengkonfigurasi kolom yang digunakan saat pelatihan 394
Mengoptimalkan solusi untuk tujuan tambahan 396
Mengoptimalkan solusi dengan konfigurasi peristiwa 403
Hyperparameters dan HPO 408
Memilih data interaksi item yang digunakan untuk pelatihan 412
Mengkloning solusi (konsol) 414
Memperbarui solusi 415
Memperbarui solusi (konsol) 416
Memperbarui solusi (AWS CLI) 417
Memperbarui solusi (AWS SDKs) 417
Membuat versi solusi secara manual 418
Membuat versi solusi (konsol) 419
Membuat versi solusi (AWS CLI) 419
Membuat versi solusi (AWS SDKs) 421
Menghentikan pembuatan versi solusi 425
Menghentikan pembuatan versi solusi (konsol) 425
Menghentikan pembuatan versi solusi (AWS CLI) 426

Menghentikan pembuatan versi solusi (AWS SDKs) 427
Mengevaluasi versi solusi 428
Mengambil metrik versi solusi 429
Definisi metrik 432
Contoh 438
Sumber daya tambahan 438
Membuat kampanye 439
Pembaruan kampanye otomatis 440
TPS minimum yang disediakan 441
Metadata item dalam rekomendasi 441
Membuat kampanye (konsol) 443
Membuat kampanye (AWS CLI) 444
Membuat kampanye (AWS SDKs) 446
Memperbarui kampanye 448
Memperbarui kampanye (konsol) 449
Memperbarui kampanye (AWS CLI) 450
Memperbarui kampanye (AWS SDKs) 450
Mendapatkan rekomendasi 453
Skor rekomendasi 453
Rekomendasi item waktu nyata 454
Cara kerja penilaian rekomendasi (sumber daya khusus) 455
Alasan rekomendasi dengan User-Personalization-v 2 456
Mendapatkan rekomendasi item waktu nyata 457
Mendapatkan metadata item dengan rekomendasi 461
Mempromosikan item 463
Rekomendasi tindakan waktu nyata 473
Cara kerja penilaian rekomendasi tindakan 474
Mendapatkan rekomendasi tindakan (konsol) 474
Mendapatkan rekomendasi tindakan (AWS CLI) 475
Mendapatkan rekomendasi tindakan (AWS SDKs) 475
Mendapatkan peringkat yang dipersonalisasi (sumber daya khusus) 476
Cara kerja penilaian peringkat yang dipersonalisasi 476
Mendapatkan peringkat yang dipersonalisasi (konsol) 477
Mendapatkan peringkat yang dipersonalisasi (AWS CLI) 478
Mendapatkan peringkat yang dipersonalisasi (AWS SDKs) 479
Notebook sampel 483

Menggunakan metadata kontekstual 483
Mendapatkan rekomendasi menggunakan metadata kontekstual (Python SDK)AWS 484
Mendapatkan rekomendasi item batch 484
Alur kerja Batch 485
Pedoman dan persyaratan 486
Penilaian alur kerja Batch 486
Rekomendasi Batch dengan tema 487
Mempersiapkan data input untuk rekomendasi batch 490
Membuat pekerjaan inferensi batch 494
Contoh keluaran pekerjaan inferensi Batch 504
Mendapatkan segmen pengguna batch 506
Pedoman dan persyaratan untuk mendapatkan segmen pengguna 507
Mempersiapkan data input untuk segmen pengguna 508
Membuat pekerjaan segmen batch 511
Contoh format output pekerjaan segmen Batch 518
Hasil penyaringan 520
Ekspresi Filter 521
Pedoman dan persyaratan 523
Filter struktur ekspresi dan elemen 524
Contoh ekspresi filter 526
Memfilter rekomendasi waktu nyata 532
Memfilter rekomendasi waktu nyata (konsol) 533
Memfilter rekomendasi waktu nyata ()AWS CLI 539
Memfilter rekomendasi waktu nyata ()AWS SDKs 541
Memfilter rekomendasi batch dan segmen pengguna (sumber daya khusus) 546
Memberikan nilai filter di JSON masukan Anda 547
Memfilter alur kerja batch (konsol) 548
Memfilter alur kerja batch ()AWS SDKs 549
Merekam acara 550
Bagaimana peristiwa waktu nyata memengaruhi rekomendasi 551
Merekam peristiwa interaksi item 551
Persyaratan untuk merekam acara interaksi item dan melatih model 552
Membuat pelacak peristiwa interaksi item 553
Merekam peristiwa interaksi item tunggal 556
Merekam beberapa peristiwa interaksi item dengan data nilai acara 560
Merekam peristiwa interaksi item dengan data tayangan 563

Metrik peristiwa dan laporan atribusi 565
Merekam peristiwa interaksi aksi 566
Persyaratan untuk merekam peristiwa interaksi tindakan 567
ID pelacak peristiwa interaksi aksi 568
Merekam peristiwa interaksi aksi tunggal 568
Merekam beberapa peristiwa interaksi aksi 570
Merekam acara untuk pengguna anonim 571
Membangun riwayat acara berkelanjutan untuk pengguna anonim 572
Layanan pelacakan acara pihak ketiga 572
Contoh implementasi 573
Mempertahankan relevansi rekomendasi 574
Menjaga set data saat ini 574
Mempertahankan rekomendasi domain 574
Mempertahankan solusi khusus 575
Memperbarui data dalam kumpulan data setelah pelatihan 577
Bagaimana data baru memengaruhi rekomendasi waktu nyata 578
Interaksi baru 578
Item baru 579
Pengguna baru 580
Tindakan baru 581
Bagaimana data baru memengaruhi rekomendasi batch (sumber daya khusus) 581
Interaksi baru 582
Pengguna baru 582
Item baru 582
Mengganti skema dataset 583
Pedoman dan persyaratan 583
Mengganti skema kumpulan data (konsol) 584
Mengganti skema () kumpulan data AWS CLI 585
Mengganti skema () kumpulan data AWS SDKs 586
Mengekspor data 587
Persyaratan izin pekerjaan ekspor dataset 588
Kebijakan peran layanan untuk mengekspor kumpulan data 588
Kebijakan bucket Amazon S3 untuk mengekspor kumpulan data 589
Membuat pekerjaan ekspor kumpulan data di Amazon Personalize 590
Membuat pekerjaan ekspor kumpulan data (konsol) 590
Membuat pekerjaan ekspor dataset ()AWS CLI 591

Membuat pekerjaan ekspor dataset ()AWS SDKs 592
Menghapus sumber daya 597
Pedoman untuk menghapus sumber daya 597
Pesanan yang disarankan untuk penghapusan sumber daya 599
Menghapus pengguna 600
Pedoman dan persyaratan 600
Mempersiapkan daftar pengguna untuk dihapus 601
Membuat pekerjaan penghapusan data 602
Menghapus dataset 606
Menghapus dataset (konsol) 607
Menghapus dataset ()AWS CLI 607
Menghapus dataset ()AWS SDKs 607
Mengukur dampak rekomendasi 609
Mengukur dampak rekomendasi dengan atribusi metrik 610
Pedoman dan persyaratan 611
Membuat atribusi metrik 614
Memperbarui atribusi metrik 621
Menghapus atribusi metrik 628
Melihat grafik data metrik di CloudWatch 630
Menerbitkan laporan atribusi metrik ke Amazon S3 631
Mengukur dampak rekomendasi dengan A/B pengujian 636
A/B menguji praktik terbaik 637
Pengujian A/B dengan CloudWatch Terbukti 638
Personalisasi hasil pencarian dari OpenSearch 642
Contoh kasus penggunaan 643
Cara kerja plugin Amazon Personalisasi Penelusuran Ranking 643
Informasi tambahan 644
Persyaratan plugin 645
Mempersonalisasi hasil dari Amazon Service OpenSearch 646
Menyiapkan izin 647
Menginstal plugin 653
Membuat pipa 653
Menerapkan plugin 655
Membandingkan hasil 657
Memantau plugin 659
Mempersonalisasi hasil dari open source Open Search 660

Menyiapkan izin ... 661
Menginstal plugin secara manual pada OpenSearch cluster yang ada ... 663
Membuat cluster baru dan menginstal plugin dengan skrip ... 664
Membuat pipa ... 665
Menerapkan plugin ... 666
Membandingkan hasil ... 668
Memantau plugin ... 668
Bidang peringkat pencarian yang dipersonalisasi ... 669
Contoh metrik pipa ... 670
Pemberian tag pada sumber daya ... 673
Pedoman dan persyaratan ... 674
Informasi tambahan ... 674
Menambahkan tag ke Amazon Personalisasi sumber daya ... 675
Menambahkan tag (konsol) ... 675
Menambahkan tag (AWS CLI) ... 676
Menambahkan tag (AWS SDKs) ... 677
Menghapus tag dari Amazon Personalisasi sumber daya ... 680
Menghapus tag (konsol) ... 680
Menghapus tag (AWS CLI) ... 681
Menghapus tag (AWS SDKs) ... 681
Menggunakan tag dalam kebijakan IAM ... 681
Pertanyaan umum ... 684
Impor dan manajemen data ... 684
Membuat solusi khusus dan versi solusi ... 685
Penerapan model (kampanye khusus) ... 686
Rekomendasi ... 686
Rekomendasi penyaringan ... 687
Pesan kesalahan umum ... 689
Impor dan manajemen data ... 689
Membuat solusi dan versi solusi (sumber daya khusus) ... 690
Penerapan model (kampanye khusus) ... 691
Rekomendasi (Grup dataset domain) ... 691
Rekomendasi ... 691
Rekomendasi penyaringan ... 692
Menentukan sumber daya dengan AWS CloudFormation ... 693
Amazon Personalisasi dan templat AWS CloudFormation ... 693

Contoh AWS CloudFormation template untuk Amazon Personalisasi sumber daya 694
CreateDatasetGroup 694
CreateDataset 695
CreateSchema 696
CreateSolution 697
Pelajari lebih lanjut tentang AWS CloudFormation 697
Contoh kode 699
Amazon Personalize 700
Hal-hal mendasar 701
Kejadian Amazon Personalize 752
Hal-hal mendasar 753
Waktu Aktif Amazon Personalize 760
Hal-hal mendasar 761
Keamanan 770
Perlindungan data 771
Enkripsi data di Amazon Personalisasi 772
Identity and Access Management 773
Audiens 773
Mengautentikasi dengan identitas 774
Mengelola akses menggunakan kebijakan 778
Cara Amazon Personalize bekerja dengan IAM 780
Pencegahan "confused deputy" lintas layanan 788
Contoh kebijakan berbasis identitas 789
Pemecahan Masalah 795
Pemantauan CloudWatch dengan 797
Menggunakan CloudWatch metrik untuk Amazon Personalize 797
Mengakses metrik Amazon Personalisasi 798
Membuat alarm 799
Aplikasi pemantauan 801
CloudWatch metrik untuk Amazon Personalisasi 801
Logging Amazon Personalisasi panggilan API dengan AWS CloudTrail 806
Amazon Personalisasi informasi di CloudTrail 806
Contoh: Amazon Personalisasi entri file log 807
Validasi kepatuhan 808
Ketahanan 809
Keamanan infrastruktur 809

Titik akhir VPC (AWS PrivateLink) 810
Membuat titik akhir VPC antarmuka untuk Amazon Personalisasi 811
Membuat kebijakan titik akhir VPC untuk Amazon Personalize 811
Titik akhir dan kuota 813
Amazon Personalisasi titik akhir dan wilayah 813
Kepatuhan 813
Kuota layanan 813
Meminta peningkatan kuota 823
Referensi API 824
Tindakan 824
Amazon Personalize 827
Kejadian Amazon Personalize 1083
Waktu Aktif Amazon Personalize 1098
Tipe Data 1116
Amazon Personalize 1119
Kejadian Amazon Personalize 1281
Waktu Aktif Amazon Personalize 1297
Kesalahan Umum 1302
Parameter Umum 1304
Riwayat dokumen 1307
..... mcccxxvi

Apa itu Amazon Personalize?

Amazon Personalize adalah layanan pembelajaran mesin yang dikelola sepenuhnya yang menggunakan data Anda untuk menghasilkan rekomendasi item bagi pengguna Anda. Hal ini juga dapat menghasilkan segmen pengguna berdasarkan afinitas pengguna untuk item tertentu atau metadata item.

Kasus penggunaan umum meliputi yang berikut:

- Mempersonalisasi aplikasi streaming video — Anda dapat menggunakan sumber daya Amazon Personalisasi yang telah dikonfigurasi sebelumnya atau dapat disesuaikan untuk menambahkan beberapa jenis rekomendasi video yang dipersonalisasi ke aplikasi streaming Anda. Misalnya, Pilihan teratas untuk Anda, Lebih suka X dan Rekomendasi video paling populer.
- Menambahkan rekomendasi produk ke aplikasi e-niaga — Anda dapat menggunakan sumber daya Amazon Personalisasi yang telah dikonfigurasi sebelumnya atau dapat disesuaikan untuk menambahkan beberapa jenis rekomendasi produk yang dipersonalisasi ke aplikasi ritel Anda. Misalnya, Direkomendasikan untuk Anda, Sering dibeli bersama dan Pelanggan yang melihat X juga melihat rekomendasi produk.
- Menambahkan rekomendasi tindakan terbaik berikutnya secara real-time ke aplikasi Anda — Anda dapat menggunakan sumber daya Amazon Personalisasi yang dapat disesuaikan untuk merekomendasikan tindakan yang kemungkinan besar akan diambil pengguna berdasarkan perilaku mereka. Misalnya, Anda dapat menambahkan rekomendasi waktu nyata untuk mendaftar di program loyalitas, mengunduh aplikasi seluler, atau mendaftar untuk email promosi.
- Membuat email yang dipersonalisasi - Anda dapat menggunakan sumber daya Amazon Personalisasi yang dapat disesuaikan untuk menghasilkan rekomendasi batch untuk semua pengguna di daftar email. Kemudian Anda dapat menggunakan AWS layanan atau layanan pihak ketiga untuk mengirim email yang dipersonalisasi kepada pengguna yang merekomendasikan item di katalog Anda.
- Membuat kampanye pemasaran yang ditargetkan — Anda dapat menggunakan Amazon Personalize untuk menghasilkan segmen pengguna yang kemungkinan besar akan berinteraksi dengan item di katalog Anda. Kemudian Anda dapat menggunakan AWS layanan atau layanan pihak ketiga untuk membuat kampanye pemasaran bertarget yang mempromosikan item yang berbeda ke segmen pengguna yang berbeda.
- Mempersonalisasi hasil penelusuran — Anda dapat menggunakan sumber daya Amazon Personalisasi yang dapat disesuaikan untuk mempersonalisasi hasil penelusuran bagi pengguna

Anda. Misalnya, Amazon Personalize dapat memberi peringkat ulang hasil penelusuran yang Anda hasilkan. OpenSearch

Untuk sebagian besar kasus penggunaan, Amazon Personalize menghasilkan rekomendasi terutama berdasarkan data interaksi item. Data interaksi item berasal dari pengguna yang berinteraksi dengan item di katalog Anda. Misalnya, pengguna mengklik item yang berbeda. Data interaksi item Anda dapat berasal dari catatan interaksi massal historis Anda dalam file CSV, dan peristiwa real-time dari pengguna Anda saat mereka berinteraksi dengan katalog Anda. Dalam beberapa kasus, Amazon Personalize juga menggunakan data dari item dan pengguna seperti genre, harga, atau jenis kelamin. Dan untuk skenario tindakan terbaik berikutnya, ia menggunakan tindakan dan data interaksi tindakan.

Saat mengimpor data massal, Anda dapat menggunakan Amazon SageMaker AI Data Wrangler untuk mengimpor data dari 40+ sumber dan menyiapkannya untuk Amazon Personalisasi. Untuk informasi selengkapnya, lihat Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler.

Amazon Personalize mencakup operasi API untuk personalisasi waktu nyata, dan operasi batch untuk rekomendasi massal dan segmen pengguna. Anda dapat memulai dengan cepat dengan rekomendasi yang dioptimalkan kasus penggunaan untuk domain bisnis Anda, atau Anda dapat membuat sumber daya kustom yang dapat dikonfigurasi sendiri.

Topik

- Harga untuk Amazon Personalisasi
- Panduan untuk Amazon Personalisasi pengguna pertama kali
- AWS Layanan dan solusi terkait
- Layanan pihak ketiga
- Pelajari selengkapnya
- Menggunakan Amazon Personalize dengan SDK AWS

Harga untuk Amazon Personalisasi

Dengan Amazon Personalize, tidak ada biaya minimum dan tidak ada komitmen di muka. Tingkat AWS Gratis menyediakan kuota bulanan hingga 20 GB pemrosesan data per AWS wilayah yang tersedia, hingga 100 jam waktu pelatihan per AWS wilayah yang memenuhi syarat, dan hingga 180.000 permintaan rekomendasi. Tingkat gratis berlaku untuk dua bulan pertama penggunaan.

Untuk daftar lengkap biaya dan harga, lihat Harga Amazon Personalisasi.

Panduan untuk Amazon Personalisasi pengguna pertama kali

Jika Anda adalah pengguna pertama kali Amazon Personalize, sumber daya berikut dapat membantu Anda memulai.

Topik

- Menemukan Amazon Personalisasi dengan Mesin Film Ajaib
- Menavigasi materi memulai dalam panduan ini

Menemukan Amazon Personalisasi dengan Mesin Film Ajaib

The Magic Movie Machine adalah pengalaman belajar interaktif. Ini membantu Anda menemukan fitur Amazon Personalisasi dan mempelajari lebih lanjut tentang membuat rekomendasi. Untuk pengantar singkat, lihat video di bawah ini. Kemudian coba Mesin Film Ajaib.

Memulai dengan Amazon Personalize

Menavigasi materi memulai dalam panduan ini

Tinjau bagian berikut untuk memulai dengan Amazon Personalize. Untuk daftar periksa yang menyediakan daftar fitur, persyaratan, dan panduan data Amazon Personalisasi, lihat. Daftar periksa kesiapan

1. Cara kerja Amazon Personalize— Bagian ini memperkenalkan alur kerja Amazon Personalize dan memandu Anda melalui langkah-langkah untuk menciptakan pengalaman yang dipersonalisasi bagi pengguna Anda. Bagian ini juga mencakup istilah umum Amazon Personalize dan definisinya. Mulailah dengan bagian ini untuk memastikan Anda memiliki pemahaman yang baik tentang Amazon Personalisasi alur kerja dan persyaratan sebelum Anda mulai mendapatkan rekomendasi.
2. Menyiapkan Amazon Personalisasi— Di bagian ini Anda mengatur Akun AWS, mengatur izin yang diperlukan untuk menggunakan Amazon Personalize, dan mengatur AWS CLI serta untuk menggunakan dan mengelola AWS SDKs Amazon Personalize.
3. Memulai tutorial— Di bagian ini Anda mulai menggunakan Amazon Personalize dengan kumpulan data film sederhana. Lengkapi tutorial ini untuk mendapatkan pengalaman langsung dengan Amazon Personalize. Anda dapat memilih untuk memulai dengan grup kumpulan data Domain atau grup kumpulan data Kustom:

- Untuk mulai membuat grup dataset Domain, selesaikan Memulai prasyarat dan kemudian mulai tutorial diMemulai grup dataset Domain.
- Untuk memulai dengan grup kumpulan data Kustom, selesaikan Memulai prasyarat dan kemudian mulai tutorial diMemulai grup dataset Domain.

Setelah menyelesaikan latihan memulai, Anda dapat mulai menyelesaikan alur kerja Amazon Personalize. Untuk langkah-langkah dan tautan ke dokumentasi, lihatAmazon Personalisasi alur kerja.

AWS Layanan dan solusi terkait

Amazon Personalize terintegrasi secara mulus dengan layanan dan solusi lain AWS . Sebagai contoh, Anda dapat:

- Gunakan Amazon SageMaker AI Data Wrangler (Data Wrangler) untuk mengimpor data dari 40+ sumber ke kumpulan data Amazon Personalize. Data Wrangler adalah fitur Amazon SageMaker AI Studio yang menyediakan end-to-end solusi untuk mengimpor, menyiapkan, mengubah, dan menganalisis data. Untuk informasi selengkapnya, lihat Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler.
- Gunakan AWS Amplify untuk merekam peristiwa interaksi item. Amplify menyertakan JavaScript pustaka untuk merekam peristiwa dari aplikasi klien web. Dan itu termasuk perpustakaan untuk merekam peristiwa dalam kode server. Untuk informasi selengkapnya, lihat Amplify Documentation.
- Otomatiskan dan jadwalkan tugas Amazon Personalisasi dengan Mempertahankan Pengalaman yang Dipersonalisasi dengan Machine Learning. Implementasi AWS Solusi ini mengotomatiskan alur kerja Amazon Personalize, termasuk impor data, pelatihan versi solusi, dan alur kerja batch.
- Gunakan Amazon CloudWatch Terbukti untuk melakukan A/B pengujian dengan rekomendasi Amazon Personalize. Untuk informasi selengkapnya, lihat Pengujian A/B dengan CloudWatch Terbukti.

Layanan pihak ketiga

Amazon Personalize bekerja dengan baik dengan berbagai layanan pihak ketiga.

- Amplitudo — Anda dapat menggunakan Amplitudo untuk melacak tindakan pengguna untuk membantu Anda memahami perilaku pengguna Anda. Untuk informasi tentang penggunaan

Amplitude dan Amazon Personalize, lihat postingan blog Partner Network (APN) AWS berikut: Mengukur Efektivitas Personalisasi dengan Amplitudo dan Amazon Personalize.

- Braze - Anda dapat menggunakan Braze untuk mengirim email yang dipersonalisasi kepada pengguna yang merekomendasikan item di katalog Anda. Braze adalah platform perpesanan terkemuka di pasar (email, push, SMS). Untuk lokakarya yang menunjukkan cara mengintegrasikan Amazon Personalize dan Braze, lihat Workshop Amazon Personalize.
- Optimizely - Anda dapat menggunakan Optimizely untuk melakukan pengujian dengan rekomendasi A/B Amazon Personalize.
- Segmen — Anda dapat menggunakan Segmen untuk mengirim data Anda ke Amazon Personalize. Untuk informasi selengkapnya tentang mengintegrasikan Segmen dengan Amazon Personalize, lihat Amazon Personalize Destination.

Untuk mengetahui daftar lengkap mitra, lihat Amazon Personalisasi Partner.

Pelajari selengkapnya

Sumber daya berikut memberikan informasi tambahan tentang Amazon Personalize:

- Untuk referensi cepat guna membantu menentukan apakah Amazon Personalize sesuai dengan kasus penggunaan Anda, lihat Amazon Personalize Cheat Sheet di repositori Amazon Personalize samples.
- Untuk serangkaian video tentang cara menggunakan Amazon Personalize, lihat Amazon Personalize Deep Dive Video Series yang ditemukan di. YouTube
- Untuk tutorial mendalam dan contoh kode, lihat amazon-personalize-samples GitHub repositori.

Menggunakan Amazon Personalize dengan SDK AWS

AWS kit pengembangan perangkat lunak (SDKs) tersedia untuk banyak bahasa pemrograman populer. Setiap SDK menyediakan API, contoh kode, dan dokumentasi yang memudahkan developer untuk membangun aplikasi dalam bahasa pilihan mereka.

Dokumentasi SDK	Contoh kode
AWS SDK untuk C++	AWS SDK untuk C++ contoh kode

Dokumentasi SDK	Contoh kode
AWS CLI	AWS CLI contoh kode
AWS SDK untuk Go	AWS SDK untuk Go contoh kode
AWS SDK untuk Java	AWS SDK untuk Java contoh kode
AWS SDK untuk JavaScript	AWS SDK untuk JavaScript contoh kode
AWS SDK untuk Kotlin	AWS SDK untuk Kotlin contoh kode
AWS SDK untuk .NET	AWS SDK untuk .NET contoh kode
AWS SDK untuk PHP	AWS SDK untuk PHP contoh kode
Alat AWS untuk PowerShell	Alat AWS untuk PowerShell contoh kode
AWS SDK untuk Python (Boto3)	AWS SDK untuk Python (Boto3) contoh kode
AWS SDK untuk Ruby	AWS SDK untuk Ruby contoh kode
AWS SDK for Rust	AWS SDK for Rust contoh kode
AWS SDK untuk SAP ABAP	AWS SDK untuk SAP ABAP contoh kode
AWS SDK for Swift	AWS SDK for Swift contoh kode

Ketersediaan contoh

Tidak dapat menemukan apa yang Anda butuhkan? Minta contoh kode menggunakan tautan Berikan umpan balik di bagian bawah halaman ini.

Amazon Personalisasi dan AI generatif

Amazon Personalize bekerja dengan baik dengan kecerdasan buatan generatif (AI generatif). Amazon Personalize Content Generator, dengan bantuan AI generatif, dapat menambahkan tema menarik ke rekomendasi batch untuk item terkait. Content Generator adalah kemampuan AI generatif yang dikelola oleh Amazon Personalize.

Anda juga dapat menggunakan rekomendasi Amazon Personalize untuk mengintegrasikan Amazon Personalize dengan alur kerja AI generatif Anda dan meningkatkan pengalaman pengguna Anda. Misalnya, Anda dapat menambahkan rekomendasi ke permintaan AI generatif untuk membuat konten pemasaran yang disesuaikan dengan minat masing-masing pengguna Anda. Anda juga dapat membuat ringkasan singkat untuk konten yang direkomendasikan, atau merekomendasikan produk atau konten melalui bot obrolan.

Video berikut menunjukkan bagaimana Anda dapat meningkatkan rekomendasi dengan Amazon Personalize dan Generative AI.

Tingkatkan Rekomendasi dengan Amazon Personalize dan Generative AI

Fitur Amazon Personalize berikut menggunakan AI generatif atau dapat membantu Anda membangun solusi AI generatif yang membuat konten yang dipersonalisasi. Untuk contoh notebook Jupyter yang menunjukkan cara menggunakan Amazon Personalize dengan AI generatif, lihat Generative AI with Amazon Personalize in the Amazon Personalize samples repository.

Topik

- Rekomendasi dengan tema dari Content Generator
- Metadata rekomendasi
- LangChain Kode pra-konfigurasi untuk personalisasi

Rekomendasi dengan tema dari Content Generator

Amazon Personalize Content Generator dapat menambahkan tema deskriptif ke rekomendasi batch. Content Generator adalah kemampuan AI generatif yang dikelola oleh Amazon Personalize.

Saat Anda mendapatkan rekomendasi batch dengan tema, Amazon Personalize Content Generator menambahkan tema deskriptif untuk setiap set item serupa. Misalnya, jika Anda mendapatkan rekomendasi item serupa untuk item makanan sarapan, Amazon Personalize mungkin menghasilkan tema seperti Rise and shine atau Morning Essentials. Anda dapat menggunakan tema untuk

mengganti judul korsel generik, seperti Sering dibeli bersama. Atau Anda dapat memasukkan tema dalam email promosi atau kampanye pemasaran untuk opsi menu baru.

Untuk menghasilkan tema, Anda mengimpor data ke dalam interaksi Item dan kumpulan data Item, membuat solusi khusus dengan resep Item Serupa, dan menghasilkan rekomendasi batch. Data item Anda harus menyertakan deskripsi item dan informasi judul. Deskripsi dan judul item terperinci membantu Content Generator membuat tema yang lebih akurat dan menarik.

- Untuk informasi tentang alur kerja Amazon Personalize, lihat. Amazon Personalisasi detail alur kerja
- Untuk informasi tentang rekomendasi batch, lihat Mendapatkan rekomendasi item batch atauMendapatkan segmen pengguna batch.
- Untuk informasi tentang membuat rekomendasi item dengan tema, lihatRekomendasi Batch dengan tema dari Content Generator.

Metadata rekomendasi

Saat mendapatkan rekomendasi, Anda dapat memiliki metadata pengembalian Amazon Personalize tentang setiap item yang direkomendasikan dari kumpulan data Item Anda. Anda dapat menambahkan metadata ini, bersama dengan rekomendasi Amazon Personalize, ke permintaan AI generatif Anda untuk menghasilkan konten yang lebih menarik.

Misalnya, Anda mungkin menggunakan AI generatif untuk membuat email pemasaran. Anda dapat menggunakan rekomendasi Amazon Personalize dan metadatanya, seperti genre film, sebagai bagian dari rekayasa cepat untuk AI generatif. Dengan petunjuk yang dipersonalisasi, Anda dapat menggunakan AI generatif untuk membuat email pemasaran yang menarik yang disesuaikan dengan minat masing-masing pelanggan Anda.

Untuk mendapatkan metadata rekomendasi, pertama-tama Anda menyelesaikan alur kerja Amazon Personalize untuk mengimpor data dan membuat domain atau sumber daya khusus. Saat Anda membuat pemberi rekomendasi Amazon Personalize atau kampanye, aktifkan opsi untuk menyertakan metadata dalam rekomendasi. Saat Anda mendapatkan rekomendasi, Anda dapat menentukan kolom data item mana yang ingin Anda sertakan.

- Untuk informasi tentang alur kerja Amazon Personalize, lihat. Amazon Personalisasi detail alur kerja
- Untuk informasi tentang mengaktifkan metadata untuk pemberi rekomendasi, lihat Mengaktifkan metadata dalam rekomendasi (sumber daya domain).

- Untuk informasi tentang mengaktifkan metadata untuk kampanye, lihat Mengaktifkan metadata dalam rekomendasi (sumber daya khusus).
- Untuk informasi selengkapnya tentang cara menggunakan Amazon Personalize dengan AI generatif untuk membuat kampanye pemasaran, lihat Meningkatkan solusi pemasaran Anda dengan Amazon Personalize dan Generative AI.

LangChain Kode pra-konfigurasi untuk personalisasi

LangChain adalah kerangka kerja untuk mengembangkan aplikasi yang didukung oleh model bahasa. Ini fitur kode yang dibangun untuk Amazon Personalize. Anda dapat menggunakan kode ini untuk mengintegrasikan rekomendasi Amazon Personalize dengan solusi AI generatif Anda.

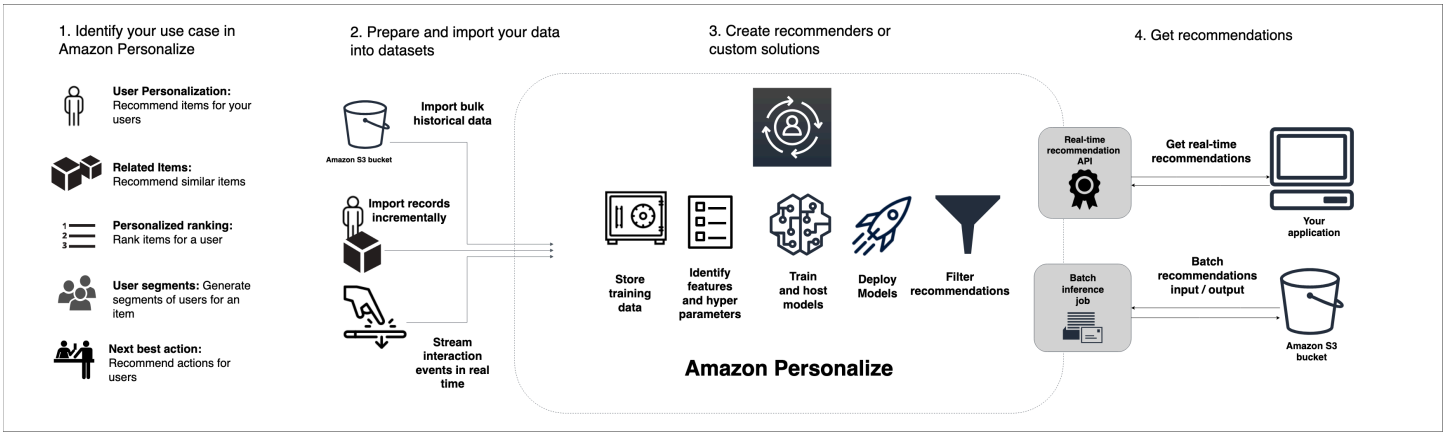
Misalnya, Anda dapat menggunakan kode berikut untuk menambahkan rekomendasi Amazon Personalisasi bagi pengguna ke rantai Anda.

```
from aws_langchain import AmazonPersonalize
from aws_langchain import AmazonPersonalizeChain
from langchain.llms.bedrock import Bedrock

recommender_arn="RECOMMENDER ARN"

bedrock_llm = Bedrock(model_id="anthropic.claude-v2", region_name="us-west-2")
client=AmazonPersonalize(credentials_profile_name="default",region_name="us-
west-2",recommender_arn=recommender_arn)
# Create personalize chain
# Use return_direct=True if you do not want summary
chain = AmazonPersonalizeChain.from_llm(
    llm=bedrock_llm,
    client=client,
    return_direct=False
)
response = chain({'user_id': '1'})
print(response)
```

- Untuk informasi tentang memulai LangChain, lihat Pendahuluan dalam LangChain dokumentasi.
- Untuk informasi tentang penggunaan LangChain kode yang dibuat untuk Amazon Personalize, termasuk contoh kode lanjutan lainnya, lihat Amazon LangChain Personalisasi ekstensi di AWS repositori sampel.

Cara kerja Amazon Personalize

Amazon Personalize menggunakan data Anda untuk melatih model rekomendasi berbasis domain atau yang dapat disesuaikan. Anda menggunakan API rekomendasi pribadi di aplikasi Anda untuk meminta rekomendasi waktu nyata. Amazon Personalize juga mendukung alur kerja batch untuk mendapatkan rekomendasi item dan segmen pengguna.



Topik

- Amazon Personalisasi detail alur kerja
- Amazon Personalisasi istilah
- Jenis data yang dapat digunakan Amazon Personalize

Amazon Personalisasi detail alur kerja

Alur kerja Amazon Personalize adalah sebagai berikut. Untuk daftar periksa yang menyediakan daftar fitur, persyaratan, dan panduan data Amazon Personalisasi, lihat. Daftar periksa kesiapan

1. Cocokkan kasus penggunaan Anda dengan Amazon Personalisasi sumber daya — Amazon Personalisasi menampilkan sumber daya berbasis domain dan sumber daya khusus yang dikonfigurasi untuk berbagai kasus. Jika Anda mencocokkan kasus penggunaan dengan sumber daya Amazon Personalize, perhatikan persyaratan datanya. Setelah Anda memilih kasus penggunaan atau resep, informasi ini dapat membantu saat Anda menyiapkan data Anda.
2. Siapkan data pelatihan Anda — Berdasarkan kasus penggunaan domain atau persyaratan data resep khusus, siapkan data pelatihan massal Anda dalam file CSV. Bergantung pada kasus penggunaan atau resep Anda, Amazon Personalize dapat menggunakan data interaksi item, item, pengguna, tindakan, dan interaksi tindakan. Jika tidak memiliki data massal, Anda dapat

menggunakan operasi impor individual untuk mengumpulkan data dan streaming peristiwa hingga memenuhi persyaratan pelatihan Amazon Personalize dan persyaratan data kasus atau resep penggunaan domain Anda.

3. Buat file JSON skema untuk data Anda - Buat file JSON skema untuk setiap jenis data yang Anda impor. File-file ini menguraikan struktur dan konten data Anda, termasuk nama kolom dan tipe datanya.
4. Buat grup kumpulan data — Grup kumpulan data adalah wadah untuk sumber daya Amazon Personalisasi. Anda dapat membuat grup kumpulan data Domain dengan sumber daya yang telah dikonfigurasi sebelumnya untuk domain VIDEO_ON_DEMAND atau ECOMMERCE. Atau Anda dapat membuat grup kumpulan data Kustom dan hanya membuat sumber daya khusus.
5. Buat skema dan kumpulan data — Skema memberi tahu Amazon Personalisasi tentang struktur data Anda dan memungkinkan Amazon Personalisasi untuk mengurai data. Dataset adalah wadah untuk melatih data di Amazon Personalize.
6. Impor data pelatihan ke dalam kumpulan data — Impor catatan interaksi interaksi, item, pengguna, tindakan, atau tindakan yang sudah disiapkan. Anda dapat mengimpor catatan dalam jumlah besar atau satu per satu.
7. Latih dan terapkan model — Untuk melatih dan menerapkan model di domain VIDEO_ON_DEMAND atau ECOMMERCE, Anda membuat rekomendasi domain. Untuk sumber daya khusus, Anda membuat solusi khusus dan versi solusi. Untuk rekomendasi real-time, Anda menerapkan versi solusi dalam kampanye.
   - Untuk informasi tentang membuat pemberi rekomendasi domain, lihatRekomendasi domain.
   - Untuk informasi tentang membuat dan menerapkan sumber daya kustom, lihatSumber daya khusus.
8. Dapatkan rekomendasi — Gunakan pemberi rekomendasi atau kampanye khusus Anda untuk mendapatkan rekomendasi. Anda dapat menggunakan filter untuk menyertakan atau mengecualikan jenis item tertentu dari rekomendasi. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna. Dengan sumber daya khusus, Anda juga bisa mendapatkan rekomendasi batch atau segmen pengguna tanpa membuat kampanye.
9. Rekam peristiwa waktu nyata — Rekam peristiwa waktu nyata saat pelanggan Anda berinteraksi dengan rekomendasi. Ini membangun data interaksi Anda dan membuat data Anda tetap segar. Dan itu memberi tahu Amazon Personalisasi tentang minat pengguna Anda saat ini, yang dapat meningkatkan relevansi rekomendasi.

Setelah Anda menyelesaikan alur kerja Amazon Personalize pertama kali, jaga agar data Anda tetap terkini, dan secara teratur latih ulang solusi kustom apa pun yang menggunakan pelatihan manual. Hal ini memungkinkan model Anda untuk belajar dari aktivitas terbaru pengguna Anda dan mempertahankan serta meningkatkan relevansi rekomendasi. Lihat informasi yang lebih lengkap di Mempertahankan relevansi rekomendasi.

Amazon Personalisasi istilah

Bagian ini memperkenalkan istilah yang digunakan di Amazon Personalize.

Topik

- Impor dan manajemen data
- Pelatihan
- Penyebaran model dan rekomendasi

Impor dan manajemen data

Istilah berikut terkait dengan mengimpor, mengekspor, dan memformat data di Amazon Personalize.

dataset tindakan

Wadah untuk metadata tentang tindakan Anda. Tindakan adalah aktivitas penghasil keterlibatan atau pendapatan yang mungkin ingin Anda rekomendasikan kepada pengguna, seperti menginstal aplikasi seluler, atau bergabung dengan program loyalitas Anda. Metadata untuk tindakan mungkin mencakup stempel waktu kedaluwarsa tindakan, nilai, data frekuensi berulang, dan metadata kategoris tindakan. Jenis data ini hanya digunakan olehNext-Best-Action resep.

kumpulan data interaksi tindakan

Wadah untuk data historis dan real-time yang Anda kumpulkan dari interaksi antara pengguna dan tindakan. Setiap interaksi tindakan terdiri dari userId, actionId, timestamp, tipe peristiwa, dan data tambahan apa pun tentang interaksi, seperti metadata kategoris. Jenis data ini hanya digunakan olehNext-Best-Action resep.

metadata kontekstual

Data interaksi yang Anda kumpulkan tentang konteks penelusuran pengguna (seperti perangkat yang digunakan atau lokasi) saat peristiwa (seperti klik) terjadi. Metadata kontekstual dapat meningkatkan relevansi rekomendasi untuk pengguna baru dan yang sudah ada.

set data

Wadah untuk data yang Anda unggah ke Amazon Personalize. Ada lima jenis kumpulan data Amazon Personalisasi: Pengguna, Item, Kumpulan data interaksi item, kumpulan data interaksi tindakan, dan Tindakan.

grup set data

Penampung untuk Amazon Personalisasi sumber daya, termasuk kumpulan data, pemberi rekomendasi domain, dan sumber daya khusus. Grup kumpulan data mengatur sumber daya Anda ke dalam koleksi independen, di mana sumber daya dari satu grup kumpulan data tidak dapat memengaruhi sumber daya dalam grup kumpulan data lainnya. Grup kumpulan data dapat berupa grup kumpulan data Domain atau grup kumpulan data Kustom.

Grup dataset domain

Grup kumpulan data yang berisi sumber daya yang telah dikonfigurasi sebelumnya untuk domain bisnis dan kasus penggunaan yang berbeda. Amazon Personalize mengelola siklus hidup model pelatihan dan penerapan. Saat membuat grup kumpulan data Domain, Anda memilih domain bisnis, mengimpor data, dan membuat rekomendasi untuk setiap kasus penggunaan Anda. Anda menggunakan pemberi rekomendasi Anda dalam aplikasi Anda untuk mendapatkan rekomendasi tentang GetRecommendations operasi.

Jika Anda memulai dengan grup kumpulan data Domain, Anda masih dapat menambahkan sumber daya khusus seperti solusi dan versi solusi yang dilatih dengan resep untuk kasus penggunaan khusus.

Grup dataset kustom

Grup kumpulan data yang hanya berisi sumber daya khusus, termasuk solusi, versi solusi, filter, kampanye, dan pekerjaan inferensi batch. Anda menggunakan kampanye untuk mendapatkan rekomendasi dengan GetRecommendations operasi. Anda mengelola siklus hidup model pelatihan dan penyebaran. Jika Anda memulai dengan grup kumpulan data Kustom, Anda tidak dapat mengaitkannya dengan domain nanti. Sebagai gantinya, buat grup dataset Domain baru.

pekerjaan ekspor dataset

Alat ekspor rekaman yang mengeluarkan catatan dalam kumpulan data ke satu atau beberapa file CSV di bucket Amazon S3. File CSV keluaran menyertakan baris header dengan nama kolom yang cocok dengan bidang dalam skema kumpulan data.

pekerjaan impor dataset

Alat impor massal yang mengisi kumpulan data Amazon Personalize Anda dengan data dari file CSV di bucket Amazon S3 Anda.

kejadian

Tindakan pengguna — seperti klik, pembelian, atau penayangan video — yang Anda rekam dan unggah ke kumpulan data interaksi Item Personalisasi Amazon. Anda mengimpor peristiwa secara massal dari file CSV, secara bertahap dengan konsol Amazon Personalize, dan secara real-time.

impresi eksplisit

Daftar item yang Anda tambahkan secara manual ke kumpulan data interaksi Item Personalisasi Amazon. Tidak seperti tayangan implisit, yang secara otomatis diperoleh Amazon Personalize dari data rekomendasi Anda, Anda memilih apa yang akan disertakan dalam tayangan eksplisit.

impresi implisit

Rekomendasi yang ditunjukkan aplikasi Anda kepada pengguna. Tidak seperti tayangan eksplisit, yang Anda tambahkan secara manual ke kumpulan data interaksi Item, Amazon Personalize secara otomatis memperoleh tayangan implisit dari data rekomendasi Anda.

data impresi

Daftar item yang Anda sajikan kepada pengguna ketika mereka berinteraksi dengan item tertentu dengan mengkliknya, menontonnya, membelinya, dan sebagainya. Amazon Personalize menggunakan data impresi untuk menghitung relevansi item baru untuk pengguna berdasarkan seberapa sering pengguna telah memilih atau mengabaikan item yang sama.

kumpulan data interaksi

Wadah untuk data historis dan real-time yang Anda kumpulkan dari interaksi antara pengguna dan item (disebut peristiwa). Data interaksi dapat mencakup data tipe peristiwa dan metadata kontekstual.

dataset item

Wadah untuk metadata tentang item Anda, seperti harga, genre, atau ketersediaan.

frekuensi berulang

Jenis metadata tindakan yang dapat Anda impor ke kumpulan data Tindakan. Data frekuensi berulang menentukan berapa hari Amazon Personalize harus menunggu untuk

merekomendasikan tindakan tertentu setelah pengguna berinteraksi dengannya, berdasarkan riwayat pengguna dalam kumpulan data interaksi Tindakan Anda.

skema

Objek JSON dalam format Apache Avro yang memberi tahu Amazon Personalisasi tentang struktur data Anda. Amazon Personalize menggunakan skema Anda untuk mengurai data Anda.

dataset pengguna

Wadah untuk metadata tentang pengguna Anda, seperti usia, jenis kelamin, atau keanggotaan loyalitas.

Pelatihan

Istilah-istilah berikut berhubungan dengan melatih model di Amazon Personalize.

item-to-item resep kesamaan (SIMS)

Resep RELATED_ITEMS yang menggunakan data dari kumpulan data Interaksi untuk membuat rekomendasi untuk item yang mirip dengan item tertentu. Resep SIMS menghitung kesamaan berdasarkan cara pengguna berinteraksi dengan item alih-alih mencocokkan metadata item, seperti harga atau warna.

item-afinitas

Resep USER_SEGMENTATION yang menggunakan data dari kumpulan data interaksi Item dan kumpulan data Item untuk membuat segmen pengguna untuk setiap item yang Anda tentukan berdasarkan kemungkinan pengguna akan berinteraksi dengan item tersebut.

item-attribute-affinity

Resep USER_SEGMENTATION yang menggunakan data dari kumpulan data interaksi Item dan kumpulan data Item untuk membuat segmen pengguna untuk setiap atribut item yang Anda tentukan berdasarkan kemungkinan pengguna akan berinteraksi dengan item dengan atribut tersebut.

Next-Best-Action resep

Resep ini menghasilkan rekomendasi real-time untuk tindakan terbaik berikutnya bagi pengguna Anda. Tindakan terbaik berikutnya untuk pengguna adalah tindakan yang kemungkinan besar akan mereka ambil. Misalnya, mendaftar di program loyalitas Anda, mengunduh aplikasi Anda,

atau mengajukan permohonan kartu kredit. Untuk informasi selengkapnya, lihat Next-Best-Action resep.

Resep peringkat-v2 yang dipersonalisasi

Resep PERSONALIZED_RANKING yang memberi peringkat kumpulan item yang Anda berikan berdasarkan tingkat minat yang diprediksi untuk pengguna tertentu. Resep ini menggunakan arsitektur berbasis transformator untuk melatih model yang belajar dari data interaksi item, metadata item, dan metadata pengguna. Gunakan resep Personalized-Ranking-v 2 untuk mempersonalisasi urutan daftar item yang dikuratori atau hasil pencarian yang dipersonalisasi untuk pengguna tertentu. Ini dapat melatih hingga 5 juta item dan menghasilkan rekomendasi yang lebih relevan dengan latensi lebih rendah dari versi sebelumnya.

resep personalized-ranking

Resep PERSONALIZED_RANKING yang memberi peringkat kumpulan item yang Anda berikan berdasarkan tingkat minat yang diprediksi untuk pengguna tertentu. Gunakan resep peringkat yang dipersonalisasi untuk mempersonalisasi urutan daftar item yang dikuratori atau hasil pencarian yang dipersonalisasi untuk pengguna tertentu.

resep popularity-count

Resep USER_PERSONALIZATION yang merekomendasikan item yang memiliki interaksi paling banyak dengan pengguna unik.

pemberi rekomendasi

Alat grup dataset Domain yang menghasilkan rekomendasi. Anda membuat pemberi rekomendasi untuk grup kumpulan data Domain dan menggunakannya dalam aplikasi Anda untuk mendapatkan rekomendasi real-time dengan API. GetRecommendations Saat membuat pemberi rekomendasi, Anda menentukan kasus penggunaan dan Amazon Personalize melatih model yang mendukung pemberi rekomendasi dengan konfigurasi terbaik untuk kasus penggunaan.

resep

Algoritme Personalisasi Amazon yang telah dikonfigurasi sebelumnya untuk memprediksi item yang akan berinteraksi dengan pengguna (untuk resep USER_PERSONALIZATION), atau menghitung item yang mirip dengan item tertentu yang diminati pengguna (untuk resep RELATED_ITEMS), atau memberi peringkat kumpulan item yang Anda berikan berdasarkan minat yang diprediksi untuk pengguna tertentu (untuk resep PERSONALIZED_RANKING).

solusi

Resep, parameter yang disesuaikan, dan model terlatih (Versi Solusi) yang digunakan Amazon Personalize untuk menghasilkan rekomendasi.

versi solusi

Model terlatih yang Anda buat sebagai bagian dari solusi di Amazon Personalize. Anda menerapkan versi solusi dalam kampanye untuk mengaktifkan API personalisasi yang Anda gunakan untuk meminta rekomendasi.

mode pelatihan

Ruang lingkup pelatihan yang akan dilakukan saat membuat versi solusi. Ada dua mode berbeda: FULL dan UPDATE. Mode PENUH membuat versi solusi yang sama sekali baru berdasarkan keseluruhan data pelatihan dari kumpulan data dalam grup kumpulan data Anda. UPDATE secara bertahap memperbarui versi solusi yang ada untuk merekomendasikan item baru yang Anda tambahkan sejak pelatihan terakhir.

> **Note**
>
> Dengan User-Personalization-v 2, Personalisasi Pengguna, atau, Amazon Next-Best-Action Personalize secara otomatis memperbarui versi solusi terbaru yang dilatih dengan mode pelatihan LENGKAP. Lihat Pembaruan otomatis.

Resep personalisasi-v2 pengguna

Resep USER_PERSONALIZATION yang merekomendasikan item yang akan berinteraksi dengan pengguna berdasarkan preferensi mereka. Resep ini menggunakan arsitektur berbasis transformator untuk melatih model yang belajar dari data interaksi item, metadata item, dan metadata pengguna. Ini dapat melatih hingga 5 juta item dan menghasilkan rekomendasi yang lebih relevan dengan latensi lebih rendah dari versi sebelumnya.

Resep Personalisasi Pengguna

Resep USER_PERSONALIZATION berbasis Hierarchical Recurrent Neural Network (HRNN) yang memprediksi item yang akan berinteraksi dengan pengguna. Resep user-personalization dapat menggunakan eksplorasi item dan data impresi untuk menghasilkan rekomendasi untuk item baru.

Penyebaran model dan rekomendasi

Istilah berikut berhubungan dengan penerapan dan penggunaan model untuk menghasilkan rekomendasi.

periode optimasi tindakan

Periode waktu yang digunakan Amazon Personalize saat memprediksi tindakan yang kemungkinan besar akan diambil pengguna. Misalnya, jika periode pengoptimalan tindakan adalah 14 hari, Amazon Personalize memprediksi tindakan yang kemungkinan besar akan dilakukan pengguna dalam 14 hari ke depan. Anda mengonfigurasi periode pengoptimalan tindakan saat Anda membuat solusi dengan Next-Best-Action resep.

pekerjaan inferensi batch

Alat yang mengimpor data input batch Anda dari bucket Amazon S3, menggunakan versi solusi Anda untuk menghasilkan rekomendasi, dan mengekspor rekomendasi ke bucket Amazon S3. Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda). Gunakan pekerjaan inferensi batch untuk mendapatkan rekomendasi untuk kumpulan data besar yang tidak memerlukan pembaruan waktu nyata.

pekerjaan segmen batch

Alat yang mengimpor data input batch Anda dari bucket Amazon S3, menggunakan versi solusi Anda untuk membuat segmen pengguna, dan mengekspor segmen pengguna ke bucket Amazon S3. Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda). Gunakan pekerjaan segmen batch dengan solusi yang didukung oleh resep USER_SEGMENTATION untuk membuat segmen pengguna berdasarkan kemungkinan pengguna akan berinteraksi dengan item atau item yang berbeda dengan atribut item yang berbeda.

kampanye

Versi solusi yang diterapkan (model terlatih) dengan kapasitas transaksi khusus yang disediakan untuk membuat rekomendasi waktu nyata bagi pengguna aplikasi Anda. Setelah membuat kampanye, Anda menggunakan operasi `getRecommendations` atau `getPersonalizedRanking` API untuk mendapatkan rekomendasi.

eksplorasi item

Dengan eksplorasi, rekomendasi mencakup beberapa item atau tindakan yang biasanya cenderung tidak direkomendasikan bagi pengguna, seperti item atau tindakan baru, item atau

tindakan dengan sedikit interaksi, atau item atau tindakan yang kurang relevan bagi pengguna berdasarkan perilaku mereka sebelumnya.

atribusi metrik

Alat yang Anda gunakan untuk mengukur dampak rekomendasi item. Atribusi metrik membuat laporan berdasarkan interaksi item dan data item yang Anda impor, dan metrik yang Anda tentukan. Misalnya, total panjang film yang ditonton oleh pengguna, atau jumlah total peristiwa klik.

rekomendasi

Daftar item yang Amazon Personalize memprediksi pengguna akan berinteraksi dengan. Bergantung pada resep Amazon Personalize yang digunakan, rekomendasi dapat berupa daftar item (resep USER_PERSONALIZATION dan resep RELATED_ITEMS), atau peringkat kumpulan item yang Anda berikan (resep PERSONALIZED_RANKING).

segmen pengguna

Daftar pengguna yang Amazon Personalize memprediksi pengguna akan berinteraksi dengan katalog Anda. Bergantung pada resep USER_SEGMENTATION yang digunakan, Anda membuat segmen pengguna berdasarkan metadata item (resep Item-Affinity) item (resep). Item-Attribute-Affinity Anda membuat segmen pengguna dengan pekerjaan segmen batch.

Jenis data yang dapat digunakan Amazon Personalize

Topik berikut memperkenalkan berbagai jenis data yang dapat Anda impor ke Amazon Personalize.

Topik

- Data interaksi
- Data barang
- Data pengguna
- Data tindakan
- Data interaksi tindakan

Data interaksi

Interaksi adalah peristiwa yang Anda rekam dan kemudian impor sebagai data pelatihan. Amazon Personalize menghasilkan rekomendasi terutama berdasarkan data interaksi. Data interaksi dapat mencakup yang berikut:

- Jenis peristiwa dan data nilai acara
- Metadata kontekstual
- Data tayangan

Anda mengimpor data interaksi ke dalam kumpulan data interaksi Item. Untuk detail selengkapnya tentang kumpulan data interaksi Item, lihat. Data interaksi item

Data barang

Metadata item yang dapat digunakan Amazon Personalize mencakup yang berikut:

- Data numerik tentang setiap item, seperti harganya.
- Metadata kategoris tentang setiap item, seperti genre atau warna item.
- Pembuatan data stempel waktu untuk setiap item.
- Metadata teks tidak terstruktur, seperti deskripsi produk atau sinopsis film.

Anda mengimpor metadata tentang item Anda ke dalam kumpulan data Item. Untuk informasi selengkapnya tentang kumpulan data Item, lihat. Metadata barang

Data pengguna

Metadata pengguna yang dapat digunakan Amazon Personalize mencakup yang berikut ini:

- Data numerik tentang setiap pengguna, seperti usia mereka.
- Metadata kategoris tentang setiap pengguna, seperti jenis kelamin atau status keanggotaan loyalitas mereka.

Anda mengimpor metadata tentang pengguna Anda ke dalam kumpulan data Pengguna. Untuk informasi selengkapnya tentang kumpulan data Pengguna, lihat. Metadata pengguna

Data tindakan

Data tindakan yang dapat digunakan Amazon Personalize mencakup hal-hal berikut:

- Nilai bisnis atau pentingnya setiap tindakan.
- Metadata kategoris untuk setiap tindakan, seperti musiman atau eksklusivitas tindakan.
- Data stempel waktu kedaluwarsa tindakan yang menentukan kapan Amazon Personalize harus berhenti merekomendasikan setiap tindakan.
- Data frekuensi berulang yang menentukan Amazon Personalize yang lama harus menunggu sebelum merekomendasikan setiap tindakan setelah pengguna berinteraksi dengannya.

Anda mengimpor data tentang tindakan Anda ke dalam kumpulan data Tindakan. Anda tidak dapat membuat sumber daya tindakan terbaik berikutnya, termasuk kumpulan data Tindakan dan Interaksi Tindakan, dalam grup kumpulan data domain. Untuk informasi selengkapnya tentang kumpulan data Tindakan, lihat. Metadata aksi

Data interaksi tindakan

Data yang dapat digunakan Amazon Personalize dari interaksi pengguna dengan tindakan termasuk yang berikut:

- Data tipe acara
- Metadata kategoris

Anda mengimpor data interaksi ke dalam kumpulan data interaksi Tindakan. Anda tidak dapat membuat sumber daya tindakan terbaik berikutnya, termasuk kumpulan data Tindakan dan Interaksi Tindakan, dalam grup kumpulan data domain. Untuk detail selengkapnya tentang kumpulan data interaksi aksi, lihat. Data interaksi aksi

Menyiapkan Amazon Personalisasi

Sebelum menggunakan Amazon Personalize, Anda harus memiliki akun Amazon Web Services (AWS) dengan pengguna administratif. Setelah mengatur izin yang diperlukan, Anda dapat mengakses Amazon Personalize melalui konsol Amazon Personalize,AWS CLI(), AWS Command Line Interface atau. AWS SDKs

Topik

- Mendaftar untuk Akun AWS
- Buat pengguna dengan akses administratif
- Wilayah dan titik akhir
- Menyiapkan izin
- Menyiapkan AWS CLI
- Menyiapkan AWS SDKs

Mendaftar untuk Akun AWS

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka https://portal.aws.amazon.com/billing/pendaftaran.
2. Ikuti petunjuk online.

   Bagian dari prosedur pendaftaran melibatkan menerima panggilan telepon atau pesan teks dan memasukkan kode verifikasi pada keypad telepon.

   Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWSdibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik keamanan terbaik, tetapkan akses administratif ke pengguna, dan gunakan hanya pengguna root untuk melakukan tugas yang memerlukan akses pengguna root.

AWS mengirimi Anda email konfirmasi setelah proses pendaftaran selesai. Kapan saja, Anda dapat melihat aktivitas akun Anda saat ini dan mengelola akun Anda dengan masuk https://aws.amazon.comke/ dan memilih Akun Saya.

Buat pengguna dengan akses administratif

Setelah Anda mendaftar Akun AWS, amankan Pengguna root akun AWS, aktifkan AWS IAM Identity Center, dan buat pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

Amankan Anda Pengguna root akun AWS

1. Masuk ke AWS Management Consolesebagai pemilik akun dengan memilih pengguna Root dan memasukkan alamat Akun AWS email Anda. Di laman berikutnya, masukkan kata sandi.

   Untuk bantuan masuk dengan menggunakan pengguna root, lihat Masuk sebagai pengguna root di AWS Sign-In Panduan Pengguna.
2. Mengaktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

   Untuk petunjuk, lihat Mengaktifkan perangkat MFA virtual untuk pengguna Akun AWS root (konsol) Anda di Panduan Pengguna IAM.

Buat pengguna dengan akses administratif

1. Aktifkan Pusat Identitas IAM.

   Untuk mendapatkan petunjuk, silakan lihat Mengaktifkan AWS IAM Identity Center di Panduan Pengguna AWS IAM Identity Center .
2. Di Pusat Identitas IAM, berikan akses administratif ke pengguna.

   Untuk tutorial tentang menggunakan Direktori Pusat Identitas IAM sebagai sumber identitas Anda, lihat Mengkonfigurasi akses pengguna dengan default Direktori Pusat Identitas IAM di Panduan AWS IAM Identity Center Pengguna.

Masuk sebagai pengguna dengan akses administratif

- Untuk masuk dengan pengguna Pusat Identitas IAM, gunakan URL masuk yang dikirim ke alamat email saat Anda membuat pengguna Pusat Identitas IAM.

  Untuk bantuan masuk menggunakan pengguna Pusat Identitas IAM, lihat Masuk ke portal AWS akses di Panduan AWS Sign-In Pengguna.

Tetapkan akses ke pengguna tambahan

1. Di Pusat Identitas IAM, buat set izin yang mengikuti praktik terbaik menerapkan izin hak istimewa paling sedikit.

   Untuk petunjuknya, lihat Membuat set izin di Panduan AWS IAM Identity Center Pengguna.
2. Tetapkan pengguna ke grup, lalu tetapkan akses masuk tunggal ke grup.

   Untuk petunjuk, lihat Menambahkan grup di Panduan AWS IAM Identity Center Pengguna.

Wilayah dan titik akhir

Titik akhir adalah URL yang merupakan titik masuk untuk layanan web. Setiap titik akhir dikaitkan secara spesifik AWS dengan wilayah. Perhatikan wilayah default konsol Amazon Personalize, Amazon Personalize, dan Amazon SDKs Personalize AWS CLI, karena semua komponen Amazon Personalize dari kampanye tertentu (kumpulan data, solusi, kampanye, pelacak peristiwa) harus dibuat di wilayah yang sama. Untuk wilayah dan titik akhir yang didukung oleh Amazon Personalize, lihat Wilayah dan titik akhir.

Menyiapkan izin

Anda harus memberikan izin kepada pengguna, grup, atau peran untuk berinteraksi dengan sumber daya Amazon Personalisasi. Dan Anda harus memberikan izin Amazon Personalize untuk mengakses sumber daya yang Anda buat di Amazon Personalize dan untuk melakukan tugas atas nama Anda.

Untuk mengatur izin

1. Berikan izin Amazon Personalize untuk mengakses sumber daya Anda di Amazon Personalisasi dan izin untuk melakukan tugas atas nama Anda. Lihat Memberikan izin Amazon Personalize untuk mengakses sumber daya Anda.
2. Berikan izin kepada pengguna, grup, atau peran Anda untuk berinteraksi dengan Amazon Personalisasi sumber daya dan meneruskan peran layanan Anda ke Amazon Personalisasi. Lihat Memberikan izin kepada pengguna untuk mengakses Amazon Personalize.
3. Ubah kebijakan kepercayaan Amazon Personalisasi peran layanan Anda sehingga mencegah masalah wakil yang membingungkan. Untuk contoh kebijakan hubungan kepercayaan,

lihatPencegahan "confused deputy" lintas layanan. Untuk informasi yang mengubah kebijakan kepercayaan peran, lihat Memodifikasi peran.

4. Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalisasi peran layanan IAM untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.
5. Selesaikan langkah-langkah Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3 untuk menggunakan kebijakan bucket IAM dan Amazon S3 untuk memberi Amazon Personalisasi akses ke sumber daya Amazon S3 Anda.

Topik

- Memberikan izin Amazon Personalize untuk mengakses sumber daya Anda
- Memberikan izin kepada pengguna untuk mengakses Amazon Personalize
- Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3
- Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS

Memberikan izin Amazon Personalize untuk mengakses sumber daya Anda

Untuk memberikan izin Amazon Personalisasi untuk mengakses sumber daya Anda, Anda membuat kebijakan IAM yang menyediakan Amazon Personalisasi akses penuh ke sumber daya Amazon Personalisasi. Atau Anda dapat menggunakan `AmazonPersonalizeFullAccess` kebijakan AWS terkelola. `AmazonPersonalizeFullAccess`memberikan izin lebih dari yang diperlukan. Sebaiknya buat kebijakan IAM baru yang hanya memberikan izin yang diperlukan. Untuk informasi selengkapnya tentang kebijakan terkelola, lihat AWS kebijakan terkelola.

Setelah membuat kebijakan, Anda membuat peran IAM untuk Amazon Personalisasi dan melampirkan kebijakan baru ke dalamnya.

Topik

- Membuat kebijakan IAM baru untuk Amazon Personalize
- Membuat peran IAM untuk Amazon Personalize

Membuat kebijakan IAM baru untuk Amazon Personalize

Buat kebijakan IAM yang menyediakan Amazon Personalisasi akses penuh ke sumber daya Amazon Personalisasi Anda.

Cara menggunakan editor kebijakan JSON untuk membuat kebijakan

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Pada panel navigasi di sebelah kiri, pilih Kebijakan.

   Jika ini pertama kalinya Anda memilih Kebijakan, akan muncul halaman Selamat Datang di Kebijakan Terkelola. Pilih Memulai.
3. Di bagian atas halaman, pilih Buat kebijakan.
4. Di bagian Editor kebijakan, pilih opsi JSON.
5. Masukkan dokumen kebijakan JSON berikut:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "personalize:*"
            ],
            "Resource": "*"
        }
    ]
}
```

6. Pilih Berikutnya.

> **Note**
>
> Anda dapat beralih antara opsi editor Visual dan JSON kapan saja. Namun, jika Anda melakukan perubahan atau memilih Berikutnya di editor Visual, IAM dapat merestrukturisasi kebijakan Anda untuk mengoptimalkannya bagi editor visual. Untuk informasi selengkapnya, lihat Restrukturisasi kebijakan dalam Panduan Pengguna IAM.

7. Pada halaman Tinjau dan buat, masukkan Nama kebijakan dan Deskripsi (opsional) untuk kebijakan yang Anda buat. Tinjau Izin yang ditentukan dalam kebijakan ini untuk melihat izin yang diberikan oleh kebijakan Anda.
8. Pilih Buat kebijakan untuk menyimpan kebijakan baru Anda.

Membuat peran IAM untuk Amazon Personalize

Untuk menggunakan Amazon Personalize, Anda harus membuat peran AWS Identity and Access Management layanan untuk Amazon Personalize. Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM. Setelah Anda membuat peran layanan untuk Amazon Personalize, berikan peran izin tambahan yang dicantumkan seperlunya. Izin peran layanan tambahan

Untuk membuat peran layanan untuk Personalize (konsol IAM)

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi konsol IAM, pilih Peran, dan lalu pilih Buat peran.
3. Untuk jenis entitas Tepercaya, pilih Layanan AWS.
4. Untuk kasus Layanan atau penggunaan, pilih Personalisasi, lalu pilih Personalisasi kasus penggunaan.
5. Pilih Berikutnya.
6. Pilih kebijakan yang Anda buat di prosedur sebelumnya.
7. (Opsional) Tetapkan batas izin. Ini adalah fitur lanjutan yang tersedia untuk peran layanan, tetapi bukan peran tertaut layanan.

    a. Buka bagian Setel batas izin, lalu pilih Gunakan batas izin untuk mengontrol izin peran maksimum.

    IAM menyertakan daftar kebijakan yang AWS dikelola dan dikelola pelanggan di akun Anda.

    b. Pilih kebijakan yang akan digunakan untuk batas izin.
8. Pilih Berikutnya.
9. Masukkan nama peran atau akhiran nama peran untuk membantu Anda mengidentifikasi tujuan peran.

> **Important**
>
> Saat Anda memberi nama peran, perhatikan hal berikut:
>
> - Nama peran harus unik di dalam diri Anda Akun AWS, dan tidak dapat dibuat unik berdasarkan kasus.
>
> Misalnya, jangan membuat peran bernama keduanya **`PRODROLE`** dan**`prodrole`**. Ketika nama peran digunakan dalam kebijakan atau sebagai bagian dari ARN, nama peran tersebut peka huruf besar/kecil, namun ketika nama peran muncul kepada pelanggan di konsol, seperti selama proses masuk, nama peran tersebut tidak peka huruf besar/kecil.
> - Anda tidak dapat mengedit nama peran setelah dibuat karena entitas lain mungkin mereferensikan peran tersebut.

10. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk peran tersebut.
11. (Opsional) Untuk mengedit kasus penggunaan dan izin untuk peran, di Langkah 1: Pilih entitas tepercaya atau Langkah 2: Tambahkan izin, pilih Edit.
12. (Opsional) Untuk membantu mengidentifikasi, mengatur, atau mencari peran, tambahkan tag sebagai pasangan nilai kunci. Untuk informasi selengkapnya tentang penggunaan tag di IAM, lihat Tag untuk AWS Identity and Access Management sumber daya di Panduan Pengguna IAM.
13. Tinjau peran lalu pilih Buat peran.

Setelah membuat peran untuk Amazon Personalize, Anda siap memberikannya akses ke bucket Amazon S3 dan kunci apa pun. AWS KMS

Izin peran layanan tambahan

Setelah Anda membuat peran dan memberinya izin untuk mengakses sumber daya Anda di Amazon Personalize, lakukan hal berikut:

1. Ubah kebijakan kepercayaan Amazon Personalisasi peran layanan Anda sehingga mencegah masalah wakil yang membingungkan. Untuk contoh kebijakan hubungan kepercayaan, lihatPencegahan "confused deputy" lintas layanan. Untuk informasi yang mengubah kebijakan kepercayaan peran, lihat Memodifikasi peran.
2. Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalisasi peran layanan IAM

untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.

Memberikan izin kepada pengguna untuk mengakses Amazon Personalize

Untuk memberi pengguna akses ke Amazon Personalize, Anda membuat kebijakan IAM yang memberikan izin untuk mengakses sumber daya Amazon Personalisasi dan meneruskan peran layanan Anda ke Amazon Personalize. Kemudian Anda menggunakan kebijakan tersebut saat menambahkan izin ke pengguna, grup, atau peran Anda.

Membuat kebijakan IAM baru untuk pengguna Anda

Buat kebijakan IAM yang menyediakan Amazon Personalisasi akses penuh ke Amazon Personalisasi sumber daya `PassRole` dan izin untuk meneruskan peran layanan Anda ke Amazon Personalize (dibuat di). Membuat peran IAM untuk Amazon Personalize

Cara menggunakan editor kebijakan JSON untuk membuat kebijakan

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Pada panel navigasi di sebelah kiri, pilih Kebijakan.

   Jika ini pertama kalinya Anda memilih Kebijakan, akan muncul halaman Selamat Datang di Kebijakan Terkelola. Pilih Memulai.
3. Di bagian atas halaman, pilih Buat kebijakan.
4. Di bagian Editor kebijakan, pilih opsi JSON.
5. Masukkan dokumen kebijakan JSON berikut:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "personalize:*"
            ],
            "Resource": "*"
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::123456789012:role/ServiceRoleName",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "personalize.amazonaws.com"
                }
            }
        }
    ]
}
```

6. Pilih Berikutnya.

> **Note**
>
> Anda dapat beralih antara opsi editor Visual dan JSON kapan saja. Namun, jika Anda melakukan perubahan atau memilih Berikutnya di editor Visual, IAM dapat merestrukturisasi kebijakan Anda untuk mengoptimalkannya bagi editor visual. Untuk informasi selengkapnya, lihat Restrukturisasi kebijakan dalam Panduan Pengguna IAM.

7. Pada halaman Tinjau dan buat, masukkan Nama kebijakan dan Deskripsi (opsional) untuk kebijakan yang Anda buat. Tinjau Izin yang ditentukan dalam kebijakan ini untuk melihat izin yang diberikan oleh kebijakan Anda.
8. Pilih Buat kebijakan untuk menyimpan kebijakan baru Anda.

Untuk hanya memberikan izin yang diperlukan untuk melakukan tugas di Amazon Personalize, ubah kebijakan sebelumnya agar hanya menyertakan tindakan yang diperlukan bagi pengguna Anda. Untuk daftar lengkap tindakan Personalisasi Amazon, lihat Tindakan, sumber daya, dan kunci kondisi untuk Amazon Personalisasi.

Menyediakan akses ke Amazon Personalize

Lampirkan kebijakan IAM baru saat Anda memberikan izin kepada pengguna Anda.

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.

- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3

Untuk memberikan Amazon Personalize akses ke bucket Amazon S3 Anda, lakukan hal berikut:

1. Jika Anda belum melakukannya, ikuti langkah-langkah Menyiapkan izin untuk menyiapkan izin agar Amazon Personalisasi dapat mengakses sumber daya Anda di Amazon Personalisasi atas nama Anda.
2. Lampirkan kebijakan ke peran layanan Amazon Personalize (lihatMembuat peran IAM untuk Amazon Personalize) yang memungkinkan akses ke bucket Amazon S3 Anda. Untuk informasi selengkapnya, lihat Melampirkan kebijakan Amazon S3 ke peran layanan Amazon Personalisasi.
3. Lampirkan kebijakan bucket ke bucket Amazon S3 yang berisi file data Anda sehingga Amazon Personalize dapat mengaksesnya. Untuk informasi selengkapnya, lihat Melampirkan kebijakan akses Amazon Personalisasi ke bucket Amazon S3 Anda.
4. Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.

> **Note**
>
> Karena Amazon Personalize tidak berkomunikasi AWS VPCs, Amazon Personalize tidak dapat berinteraksi dengan bucket Amazon S3 yang hanya mengizinkan akses VPC.

Topik

- Melampirkan kebijakan Amazon S3 ke peran layanan Amazon Personalisasi
- Melampirkan kebijakan akses Amazon Personalisasi ke bucket Amazon S3 Anda

Melampirkan kebijakan Amazon S3 ke peran layanan Amazon Personalisasi

Untuk melampirkan kebijakan Amazon S3 ke peran Amazon Personalisasi, lakukan hal berikut:

1. Masuk ke konsol IAM (https://console.aws.amazon.com/iam/).
2. Di panel navigasi, pilih Kebijakan, dan pilih Buat kebijakan.
3. Pilih tab JSON, dan perbarui kebijakan sebagai berikut. Ganti `amzn-s3-demo-bucket` dengan nama bucket Anda. Anda dapat menggunakan kebijakan berikut untuk pekerjaan impor kumpulan data atau pekerjaan penghapusan data. Jika Anda menggunakan alur kerja batch atau membuat pekerjaan ekspor kumpulan data, Amazon Personalize memerlukan izin tambahan. Lihat Kebijakan peran layanan untuk alur kerja batch atau Kebijakan bucket Amazon S3 untuk mengekspor kumpulan data.

   JSON

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
```

```
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

4. Pilih Berikutnya: Tanda. Secara opsional tambahkan tag apa pun dan pilih Tinjau.
5. Beri nama kebijakan.
6. (Opsional) Untuk Deskripsi, masukkan kalimat pendek yang menjelaskan kebijakan ini, misalnya, **`Allow Amazon Personalize to access its Amazon S3 bucket.`**
7. Pilih Buat kebijakan.
8. Di panel navigasi, pilih Peran, dan pilih peran yang Anda buat untuk Amazon Personalize. Lihat Membuat peran IAM untuk Amazon Personalize.
9. Untuk Izin, pilih Lampirkan kebijakan.
10. Untuk menampilkan kebijakan dalam daftar, ketik bagian dari nama kebijakan di kotak Filter kebijakan filter.
11. Pilih kotak centang di samping kebijakan yang Anda buat sebelumnya dalam prosedur ini.
12. Pilih Lampirkan kebijakan.

    Sebelum peran Anda siap digunakan dengan Amazon Personalize, Anda juga harus melampirkan kebijakan bucket ke bucket Amazon S3 yang berisi data Anda. Lihat Melampirkan kebijakan akses Amazon Personalisasi ke bucket Amazon S3 Anda.

Kebijakan peran layanan untuk alur kerja batch

Untuk menyelesaikan worklfow batch, Amazon Personalize memerlukan izin untuk mengakses dan menambahkan file ke bucket Amazon S3 Anda. Ikuti langkah-langkah di atas untuk melampirkan kebijakan berikut ke peran Amazon Personalize Anda. Ganti `amzn-s3-demo-bucket` dengan nama bucket Anda. Untuk informasi selengkapnya tentang alur kerja batch, lihat Mendapatkan rekomendasi item batch atauMendapatkan segmen pengguna batch.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
```

```
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Kebijakan peran layanan untuk mengekspor kumpulan data

Untuk mengekspor kumpulan data, peran layanan Amazon Personalize Anda memerlukan izin untuk menggunakan `PutObject` `ListBucket` dan Tindakan di bucket Amazon S3 Anda. Contoh kebijakan berikut memberikan Amazon `PutObject` Personalize `ListBucket` dan izin. Ganti `amzn-s3-demo-bucket` dengan nama bucket Anda dan lampirkan kebijakan ke peran layanan Anda untuk Amazon Personalize. Untuk informasi tentang melampirkan kebijakan ke peran layanan, lihatMelampirkan kebijakan Amazon S3 ke peran layanan Amazon Personalisasi.

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
```

```
    ]
}
```

Melampirkan kebijakan akses Amazon Personalisasi ke bucket Amazon S3 Anda

Amazon Personalize memerlukan izin untuk mengakses bucket S3. Anda dapat menggunakan kebijakan berikut untuk pekerjaan impor kumpulan data atau pekerjaan penghapusan data. Ganti `amzn-s3-demo-bucket` dengan nama bucket Anda. Untuk alur kerja batch, lihatKebijakan bucket Amazon S3 untuk alur kerja batch.

Untuk informasi selengkapnya tentang kebijakan bucket Amazon S3, lihat Bagaimana Cara Menambahkan Kebijakan Bucket S3? .

JSON

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "personalize.amazonaws.com"
            },
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Kebijakan bucket Amazon S3 untuk alur kerja batch

Untuk alur kerja batch, Amazon Personalize memerlukan izin untuk mengakses dan menambahkan file ke bucket Amazon S3 Anda. Lampirkan kebijakan berikut ke bucket Anda. Ganti `amzn-s3-demo-bucket` dengan nama bucket Anda.

Untuk informasi selengkapnya tentang menambahkan kebijakan bucket Amazon S3 ke bucket, lihat Bagaimana Cara Menambahkan Kebijakan Bucket S3? . Untuk informasi selengkapnya tentang alur kerja batch, lihat Mendapatkan rekomendasi item batch atauMendapatkan segmen pengguna batch.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "personalize.amazonaws.com"
            },
            "Action": [
                "s3:GetObject",
                "s3:ListBucket",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Kebijakan bucket Amazon S3 untuk mengekspor kumpulan data

Untuk mengekspor kumpulan data, Amazon Personalize memerlukan izin untuk menggunakan `PutObject` `ListBucket` dan Tindakan di bucket Amazon S3 Anda. Contoh kebijakan berikut memberikan `PutObject` prinsip `ListBucket` dan izin Amazon Personalize. Ganti `amzn-s3-`

`demo-bucket` dengan nama bucket Anda dan lampirkan kebijakan ke bucket Anda. Untuk informasi tentang menambahkan kebijakan bucket Amazon S3 ke bucket, lihat Bagaimana Cara Menambahkan Kebijakan Bucket S3? di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "personalize.amazonaws.com"
            },
            "Action": [
                "s3:PutObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS

Jika Anda menentukan kunci AWS Key Management Service (AWS KMS) saat menggunakan konsol Amazon Personalize atau APIs, atau jika Anda menggunakan AWS KMS kunci untuk mengenkripsi bucket Amazon S3, Anda harus memberikan izin Amazon Personalize untuk menggunakan kunci Anda. Untuk memberikan izin, kebijakan AWS KMS utama dan kebijakan IAM yang dilampirkan pada peran layanan Anda harus memberikan izin Amazon Personalize untuk menggunakan kunci Anda. Ini berlaku untuk membuat yang berikut di Amazon Personalize.

- Grup kumpulan data
- Pekerjaan impor kumpulan data (hanya kebijakan AWS KMS utama yang harus memberikan izin)
- Lowongan kerja ekspor dataset

- Pekerjaan inferensi Batch
- Pekerjaan segmen Batch
- Atribusi metrik

Kebijakan AWS KMS utama dan kebijakan IAM Anda harus memberikan izin untuk tindakan berikut:

- Dekripsi
- GenerateDataKey
- DescribeKey
- CreateGrant (hanya diperlukan dalam kebijakan utama)
- ListGrants

Mencabut izin AWS KMS kunci setelah membuat sumber daya dapat menyebabkan masalah saat membuat filter atau mendapatkan rekomendasi. Untuk informasi selengkapnya tentang AWS KMS kebijakan, lihat Menggunakan kebijakan utama di AWS KMS di Panduan AWS Key Management Service Pengembang. Untuk informasi tentang cara membuat kebijakan IAM, lihat Membuat kebijakan IAM di Panduan Pengguna IAM. Untuk informasi tentang melampirkan kebijakan IAM ke peran, lihat Menambahkan dan menghapus izin identitas IAM di Panduan Pengguna IAM.

Topik

- Contoh kebijakan utama
- Contoh kebijakan IAM

Contoh kebijakan utama

Contoh kebijakan utama berikut memberikan Amazon Personalize dan peran Anda izin minimum untuk operasi Amazon Personalize sebelumnya. Jika Anda menentukan kunci saat membuat grup kumpulan data dan ingin mengekspor data dari kumpulan data, kebijakan kunci Anda harus menyertakan tindakan tersebut`GenerateDataKeyWithoutPlaintext`.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "key-policy-123",
```

```
    "Statement": [
        {
            "Sid": "Allow use of the key",
            "Effect": "Allow",
            "Principal": {
            "AWS": "arn:aws:iam::111122223333:role/<personalize-role-name>",
                "Service": "personalize.amazonaws.com"
            },
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:DescribeKey",
                "kms:CreateGrant",
                "kms:ListGrants"
            ],
            "Resource": "*"
        }
    ]
}
```

Contoh kebijakan IAM

Contoh kebijakan IAM berikut memberikan peran AWS KMS izin minimum yang diperlukan untuk operasi Amazon Personalize sebelumnya. Untuk pekerjaan impor kumpulan data, hanya kebijakan AWS KMS utama yang perlu memberikan izin.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:DescribeKey",
                "kms:ListGrants"
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

Menyiapkan AWS CLI

The AWS Command Line Interface (AWS CLI) adalah alat pengembang terpadu untuk mengelola AWS layanan, termasuk Amazon Personalize. Kami menyarankan Anda menginstalnya.

1. Untuk menginstal AWS CLI, ikuti petunjuk dalam Menginstal AWS Command Line Interface di Panduan Pengguna AWS Command Line Interface Antarmuka.
2. Untuk mengkonfigurasi AWS CLI dan mengatur profil untuk memanggil AWS CLI, ikuti petunjuk dalam Mengkonfigurasi AWS CLI dalam Panduan AWS Command Line Interface Pengguna.
3. Untuk mengonfirmasi bahwa AWS CLI profil dikonfigurasi dengan benar, jalankan perintah berikut.

   ```
   aws configure --profile default
   ```

   Jika profil Anda telah dikonfigurasi dengan benar, Anda akan melihat output yang mirip dengan berikut ini.

   ```
   AWS Access Key ID [****************52FQ]:
   AWS Secret Access Key [****************xgyZ]:
   Default region name [us-west-2]:
   Default output format [json]:
   ```

4. Untuk memverifikasi bahwa AWS CLI dikonfigurasi untuk digunakan dengan Amazon Personalize, jalankan perintah berikut.

   ```
   aws personalize help
   ```

   and

   ```
   aws personalize-runtime help
   ```

   and

   ```
   aws personalize-events help
   ```

Jika AWS CLI dikonfigurasi dengan benar, Anda akan melihat daftar AWS CLI perintah yang didukung untuk Amazon Personalize, Amazon Personalize runtime, dan Amazon Personalize event.

Jika Anda mengatur AWS CLI dan tidak mengenali perintah untuk Amazon Personalisasi, perbarui. AWS CLI Untuk memperbarui AWS CLI, jalankan perintah berikut.

```
pip3 install awscli --upgrade --user
```

Untuk informasi selengkapnya, lihat Menginstal pip AWS CLI using.

Menyiapkan AWS SDKs

Unduh dan instal AWS SDKs yang ingin Anda gunakan. Panduan ini memberikan contoh untuk SDK for Python (Boto3), SDK for Java 2.x, dan SDK untuk v3. JavaScript Untuk informasi tentang lainnya AWS SDKs, lihat Alat untuk Amazon Web Services. Untuk informasi tentang cara menyiapkan Amplify, lihat Amplify documentation.

- AWS SDK untuk Python (Boto3)

  Untuk menginstal SDK for Python (Boto3), ikuti petunjuk Quickstart dalam dokumentasi Boto3.
- SDK for Java 2.x

  Untuk mempelajari cara menyiapkan SDK for Java 2.x, lihat topik Memulai dengan SDK for Java 2.x di Panduan Pengembang.AWS SDK for Java 2.x

  Untuk contoh kode untuk Amazon Personalize, lihat Amazon Personalisasi sampel kode Java di repositori contoh AWS SDK.
- AWS SDK untuk JavaScript v3

  Untuk mempelajari cara menyiapkan SDK untuk JavaScript v3, lihat Memulai AWS SDK untuk JavaScript topik di Panduan AWS SDK untuk JavaScript Pengembang.

  Untuk contoh kode untuk Amazon Personalize, lihat contoh kode Amazon Personalize untuk SDK for v3 di repositori contoh SDK JavaScript .AWS

Memulai tutorial

Bagian berikut membantu Anda mulai menggunakan Amazon Personalize dengan konsol AWS CLI Amazon Personalize, dan. AWS SDKs Tutorial menggunakan data historis yang terdiri dari 100.000 peringkat film pada 9.700 film dari 600 pengguna.

Untuk menyederhanakan tutorial:

- Kami menggunakan dataset kecil. Ini mungkin berdampak negatif pada metrik apa pun yang dihasilkan oleh sumber daya. Tutorial berfungsi sebagai pengantar alur kerja Amazon Personalize dan belum tentu menghasilkan model dengan performa tertinggi.
- Kami hanya membuat kumpulan data interaksi Item, dan mengandalkan fakta bahwa pengguna melihat film dan bukan pada apa yang mereka beri peringkat film. Ini menyederhanakan persiapan data pelatihan.
- Kami tidak merekam peristiwa interaksi pengguna langsung. Untuk informasi tentang menangkap peristiwa pengguna, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi.

Anda dapat memilih untuk memulai grup kumpulan data Domain atau grup kumpulan data Kustom:

- Grup kumpulan data domain menyediakan sumber daya yang dioptimalkan untuk berbagai kasus penggunaan berdasarkan domain Anda. Untuk mulai membuat grup dataset Domain, selesaikan Memulai prasyarat dan kemudian selesaikan tutorialnya. Memulai grup dataset Domain
- Grup kumpulan data khusus memungkinkan Anda membuat dan mengonfigurasi hanya sumber daya khusus. Untuk mulai memberikan rekomendasi film yang dipersonalisasi untuk pengguna Anda dengan sumber daya khusus dan Resep personalisasi-v2 pengguna resep, selesaikan Memulai prasyarat dan kemudian mulai tutorial diMemulai grup Dataset Kustom.

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Topik

- Memulai prasyarat
- Memulai grup dataset Domain
- Memulai grup Dataset Kustom

Memulai prasyarat

Langkah-langkah berikut adalah prasyarat untuk latihan memulai.

1. Siapkan izin agar Amazon Personalize dapat mengakses sumber daya Anda atas nama Anda. Ini melibatkan pembuatan peran layanan untuk Amazon Personalize dan memberinya akses ke Amazon Personalize resource dengan kebijakan IAM. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk mengakses sumber daya Anda.
2. Siapkan data pelatihan Anda dan unggah data ke bucket Amazon S3 Anda:
   - Untuk tutorial grup kumpulan data Domain, lihatMembuat data pelatihan (Grup dataset domain).
   - Untuk tutorial grup kumpulan data kustom, lihatMembuat data pelatihan (grup dataset kustom).
3. Berikan izin peran layanan Amazon Personalize Anda untuk mengakses sumber daya Amazon S3 Anda, seperti yang ditentukan dalam. Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3

Membuat data pelatihan (Grup dataset domain)

Untuk membuat data pelatihan, unduh, modifikasi, dan simpan data peringkat film ke bucket Amazon Simple Storage Service (Amazon S3). Kemudian berikan izin Amazon Personalize untuk membaca dari ember.

Untuk membuat data pelatihan

1. Unduh dan unzip file zip peringkat film, ml-latest-small.zip dari MovieLensbawah yang direkomendasikan untuk pendidikan dan pengembangan (F. Maxwell Harper dan Joseph A. Konstan. 2015. MovieLens Kumpulan Data: Sejarah dan Konteks. Transaksi ACM pada Sistem Cerdas Interaktif (TII) 5, 4:19:1 - 19:19. https://doi.org/10.1145/2827872).
2. Buka file `ratings.csv`. File ini berisi data interaksi untuk tutorial ini.
   a. Hapus kolom peringkat.
   b. Ganti nama `userId` dan `movieId` kolom menjadi `USER_ID` dan `ITEM_ID` masing-masing.
   c. Tambahkan kolom EVENT_TYPE dan atur nilai untuk setiap catatan. `watch` Jika Anda menggunakan Microsoft Excel, Anda dapat mengatur EVENT_TYPE untuk setiap catatan dengan memasukkan sel pertama `watch` di kolom dan kemudian mengklik dua kali sudut kanan bawah sel. Header Anda harus sebagai berikut:

USER_ID,ITEM_ID,TIMESTAMP,EVENT_TYPE

Kolom ini harus persis seperti yang ditunjukkan untuk Amazon Personalize untuk mengenali data. Beberapa baris pertama data Anda akan terlihat sebagai berikut:

```
USER_ID,ITEM_ID,TIMESTAMP,EVENT_TYPE
1,1,964982703,watch
1,3,964981247,watch
1,6,964982224,watch
1,47,964983815,watch
1,50,964982931,watch
....
....
```

Simpan file `ratings.csv`.

3. Unggah `ratings.csv` ke bucket Amazon S3 Anda. Untuk informasi selengkapnya, lihat Mengunggah file dan folder menggunakan seret dan lepas di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.
4. Berikan izin Amazon Personalize untuk membaca data di bucket. Untuk informasi selengkapnya, lihat Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3.

Membuat data pelatihan (grup dataset kustom)

Untuk membuat data pelatihan, unduh, modifikasi, dan simpan data peringkat film ke bucket Amazon Simple Storage Service (Amazon S3). Kemudian berikan izin Amazon Personalize untuk membaca dari ember.

1. Unduh dan unzip file zip peringkat film, ml-latest-small.zip dari MovieLensbawah yang direkomendasikan untuk pendidikan dan pengembangan (F. Maxwell Harper dan Joseph A. Konstan. 2015. MovieLens Kumpulan Data: Sejarah dan Konteks. Transaksi ACM pada Sistem Cerdas Interaktif (TII) 5, 4:19:1 - 19:19. https://doi.org/10.1145/2827872).
2. Buka file `ratings.csv`. File ini berisi data interaksi untuk tutorial ini.

   a. Hapus kolom peringkat.

   b. Ganti baris header dengan yang berikut:

   USER_ID,ITEM_ID,TIMESTAMP

Header ini harus persis seperti yang ditunjukkan untuk Amazon Personalize untuk mengenali data.

Simpan file `ratings.csv`.

3. Unggah `ratings.csv` ke bucket Amazon S3 Anda. Untuk informasi selengkapnya, lihat Mengunggah file dan folder menggunakan seret dan lepas di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.
4. Berikan izin Amazon Personalize untuk membaca data di bucket. Lihat informasi yang lebih lengkap di Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3.

Memulai grup dataset Domain

Dalam tutorial memulai ini, Anda membuat grup dataset Domain untuk domain VIDEO_ON_DEMAND, mengimpor data interaksi dari file CSV, dan membuat pemberi rekomendasi dengan pilihan Teratas untuk kasus penggunaan Anda. Kemudian Anda menggunakan pemberi rekomendasi untuk mendapatkan rekomendasi film yang dipersonalisasi untuk pengguna. Tutorial ini menggunakan data historis yang terdiri dari 100.000 peringkat film pada 9.700 film dari 600 pengguna.

Untuk memulai, selesaikan Memulai prasyarat dan kemudian tergantung pada bagaimana Anda ingin membuat Amazon Personalisasi sumber daya, lanjutkan ke,Memulai grup dataset Domain (konsol), Memulai grup dataset Domain (SDK for Python (Boto3))Memulai grup dataset Domain (SDK for Java 2.x), atau. Memulai grup dataset Domain (SDK untuk JavaScript v3)

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Lihat informasi yang lebih lengkap di Persyaratan untuk menghapus sumber daya Amazon Personalize.

Topik

- Memulai grup dataset Domain (konsol)
- Memulai grup dataset Domain (SDK for Java 2.x)
- Memulai grup dataset Domain (SDK for Python (Boto3))
- Memulai grup dataset Domain (SDK untuk JavaScript v3)

Memulai grup dataset Domain (konsol)

Dalam latihan ini, Anda menggunakan konsol Amazon Personalize untuk membuat grup kumpulan data Domain dan pemberi rekomendasi yang menampilkan rekomendasi film untuk pengguna tertentu.

Sebelum Anda memulai latihan ini, tinjau kembaliMemulai prasyarat.

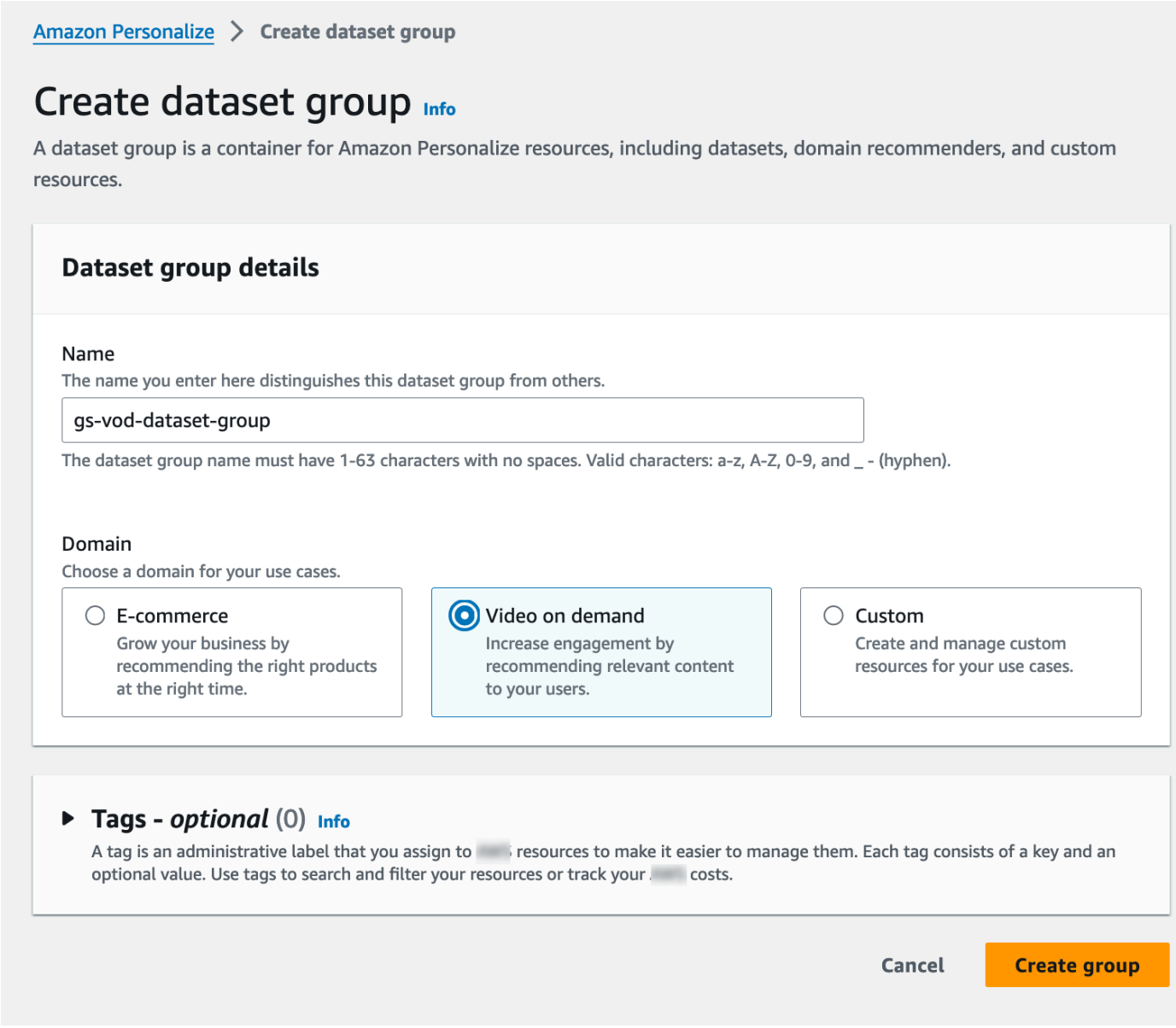
Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Langkah 1: Buat grup dataset Domain

Dalam prosedur ini Anda membuat grup kumpulan data Domain untuk domain VIDEO_ON_DEMAND, buat kumpulan data interaksi Item dengan skema default untuk domain VIDEO_ON_DEMAND, dan impor data interaksi item yang Anda buat. Membuat data pelatihan (Grup dataset domain)

Untuk membuat grup dataset Domain

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Di panel navigasi, pilih Buat grup kumpulan data.
3. Di detail grup Dataset, tentukan nama untuk grup kumpulan data Anda.
4. Untuk Domain, pilih Video sesuai permintaan. Domain yang Anda pilih menentukan skema default yang Anda gunakan saat mengimpor data. Ini juga menentukan kasus penggunaan apa yang tersedia untuk pemberi rekomendasi. Layar Anda akan terlihat mirip dengan yang berikut ini.



5. Pilih Buat grup kumpulan data. Halaman Ikhtisar muncul. Lanjut ke Langkah 2: Impor data.

Langkah 2: Impor data

Dalam prosedur ini Anda membuat kumpulan data interaksi Item dengan skema domain VIDEO_ON_DEMAND default. Kemudian Anda mengimpor data interaksi item yang Anda buatMembuat data pelatihan (Grup dataset domain).

Untuk mengimpor data

1. Pada halaman Ikhtisar, pada Langkah 1. Buat kumpulan data dan impor data, pilih Buat kumpulan data dan pilih Set data interaksi item.

2. Pilih Impor data langsung ke Amazon Personalize dataset dan pilih Berikutnya.
3. Di halaman skema interaksi item Konfigurasi, untuk nama Dataset berikan nama untuk kumpulan data interaksi Item Anda.
4. Untuk skema Dataset, pilih Buat skema domain baru dengan memodifikasi skema default yang ada untuk domain Anda dan masukkan nama untuk skema. Definisi Skema diperbarui untuk menampilkan skema default untuk domain VIDEO_ON_DEMAND. Biarkan skema tidak berubah. Layar Anda akan terlihat mirip dengan yang berikut ini.

Configure item interactions schema Info

Dataset details

Dataset name

The name you enter here can help you distinguish this dataset import job from others.

gs-interactions-ds

The dataset name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

Dataset schema

The schema you provide allows Amazon Personalize to understand and import your data.

- (●) Create a new domain schema by modifying the existing default schema for your domain
- ( ) Use an existing domain related schema

Schema name

The name you enter here can help you distinguish this schema from others.

gs-interactions-domain-schema

The schema name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

Schema definition

Verify your data structure matches the following schema.

```
{
  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
    {
      "name": "USER_ID",
      "type": "string"
    },
    {
      "name": "ITEM_ID",
      "type": "string"
    },
    {
      "name": "TIMESTAMP",
      "type": "long"
    },
```

5. Pilih Berikutnya. Halaman pekerjaan impor set data Konfigurasikan interaksi item muncul.
6. Pada halaman pekerjaan impor set data Konfigurasi interaksi item, biarkan sumber impor data tidak berubah sebagai Impor data dari S3.
7. Untuk nama pekerjaan impor Dataset, beri nama pekerjaan impor Anda.
8. Di Sumber impor data, tentukan tempat penyimpanan data Anda di Amazon Simple Storage Service (S3). Gunakan sintaks berikut:

   ```
   s3://amzn-s3-demo-bucket/<folder path>/<CSV filename>
   ```

9. Dalam peran IAM, untuk peran layanan IAM pilih Masukkan ARN peran IAM khusus dan masukkan Nama Sumber Daya Amazon (ARN) peran yang Anda buat. Membuat peran IAM untuk Amazon Personalize Layar Anda akan terlihat mirip dengan yang berikut ini.

Configure item interactions dataset import job Info

Dataset import job details

Data import source

- (●) Import data from S3
  Specify the location where your data is stored in S3.
- ( ) Incrementally import data with APIs
  Incrementally import item interactions data with the event ingestion SDK.

Dataset import job name
The name you enter here can help you distinguish this dataset import job from others.

my-dataset-import-job-name

The dataset import job name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

Data import source

Additional S3 bucket policy required
In addition to the IAM service role defined above, Amazon Personalize also requires you to add a bucket policy to the S3 bucket containing your data files so that it can process them. Follow the instructions described here to add the required bucket policy to your S3 bucket.

Data location Info
Choose the S3 location of your data.

s3://bucket/path-to-your-data/

Your file needs to be in a CSV format and reflect the schema.

IAM Role

IAM service role
Amazon Personalize requires permissions to access your S3 bucket. Choose an existing role with access or create a role in the IAM console with the AmazonPersonalizeFullAccess IAM policy attached.

Enter a custom IAM role ARN

Custom IAM role ARN

arn:aws:iam::YourAccountID:role/YourRole

After you import data from S3, you can still incrementally import data with the Amazon Personalize console, the [illegible] Command Line Interface ([illegible] CLI), or the [illegible] SDKs.

10. Pilih Mulai impor untuk mengimpor data. Halaman Ikhtisar untuk grup kumpulan data Domain Anda muncul. Perhatikan status impor di bagian Set up dataset. Ketika status `Interaction data active` dilanjutkan keLangkah 3: Buat pemberi rekomendasi.

Langkah 3: Buat pemberi rekomendasi

Dalam prosedur ini, Anda membuat pemberi rekomendasi untuk pilihan Teratas untuk kasus penggunaan Anda untuk domain VIDEO_ON_DEMAND.

Untuk membuat pemberi rekomendasi

1. Pada halaman Ikhtisar untuk grup kumpulan data Domain Anda, pada Langkah 3 pilih tab Gunakan video sesuai permintaan, lalu pilih Buat rekomendasi.
2. Pada halaman Pilih kasus penggunaan, pilih Pilihan teratas untuk Anda dan berikan nama Rekomendasi. Layar Anda akan tampak mirip dengan yang berikut ini.

Choose use case Info

You use recommenders to get recommendations for specific e-commerce use cases. Amazon Personalize trains the models backing each recommender with the optimal configurations for these use cases.

Video on demand recommenders

Video on demand use cases to create recommenders for

Amazon Personalize will create a recommender for each selected use case.

- [ ] Because you watched X

  Get recommendations for videos that other users also watched based on a video you specify.

  Recommender name

  my-recommender-name

  The recommender name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

- [ ] More like X

  Get recommendations for videos that are similar to a video you specify.

  Recommender name

  my-recommender-name

  The recommender name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

- [ ] Most popular

  Get recommendations for videos that have been watched by the most users.

  Recommender name

  my-recommender-name

  The recommender name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

- [x] Top picks for you

  Get personalized content recommendations for a user you specify.

  Recommender name

  my-recommender-name

  The recommender name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

- [ ] Trending now - *new*

  Get recommendations for videos that are trending now.

  Recommender name

  my-recommender-name

  The recommender name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

Cancel Next

3. Pilih Berikutnya.
4. Biarkan bidang pada halaman konfigurasi lanjutan tidak berubah dan pilih Berikutnya.
5. Tinjau detail pemberi rekomendasi dan pilih Buat pemberi rekomendasi untuk membuat pemberi rekomendasi Anda.

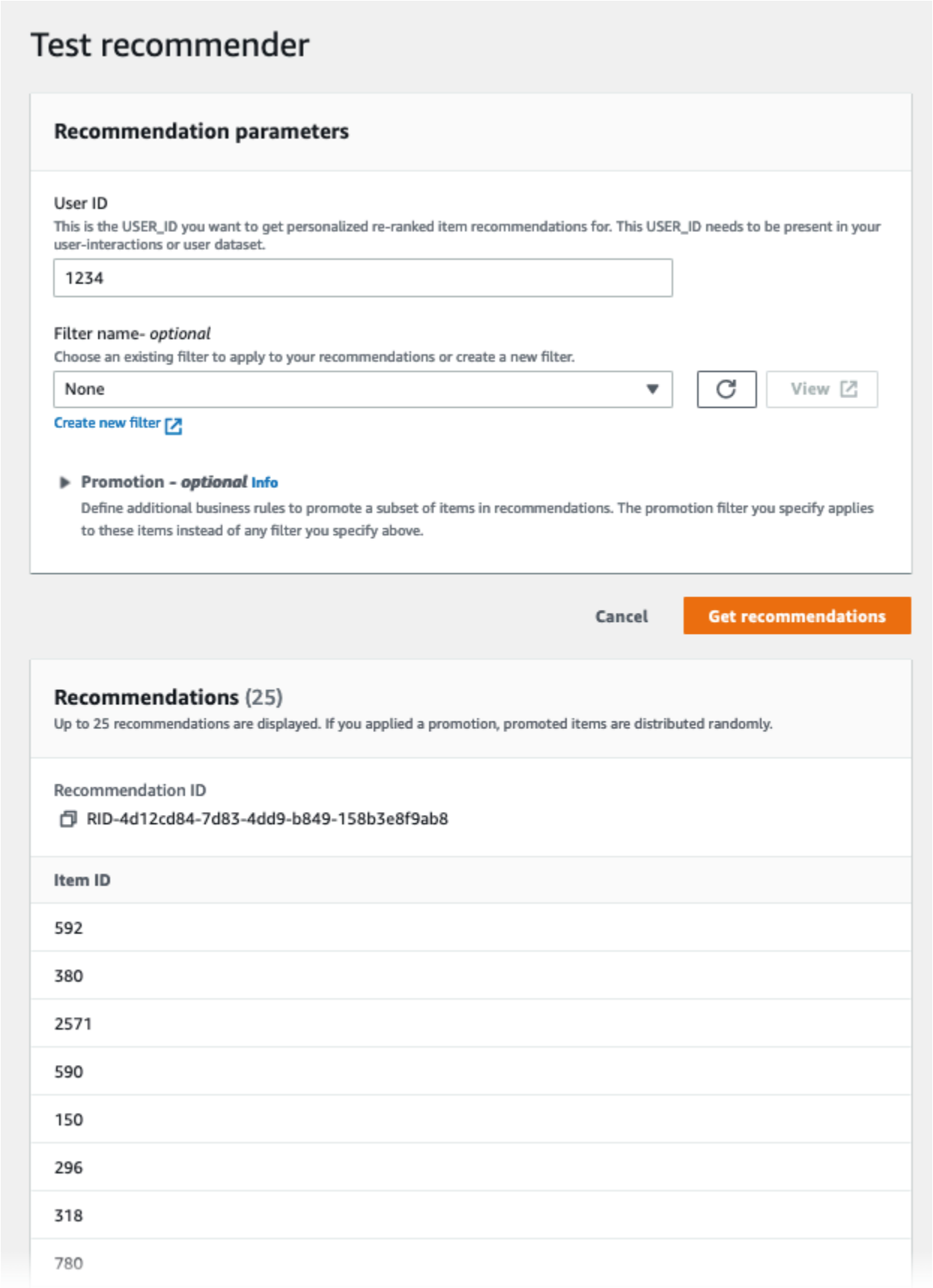
   Anda dapat memantau status masing-masing pemberi rekomendasi di halaman Rekomendasi. Ketika status pemberi rekomendasi Anda Aktif, Anda dapat menggunakannya untuk mendapatkan rekomendasi. Langkah 4: Dapatkan rekomendasi

Langkah 4: Dapatkan rekomendasi

Dalam prosedur ini Anda menggunakan pemberi rekomendasi yang Anda buat pada langkah sebelumnya untuk mendapatkan rekomendasi.

Untuk mendapatkan rekomendasi

1. Pada halaman Ringkasan untuk grup kumpulan data Domain Anda, di panel navigasi pilih Rekomendasi.
2. Pada halaman Rekomendasi, pilih pemberi rekomendasi Anda.
3. Di kanan atas, pilih Test.
4. Dalam parameter Rekomendasi, masukkan ID pengguna. Tidak perlu mengubah bidang lain.
5. Pilih Dapatkan rekomendasi. Tabel yang berisi 25 item direkomendasikan teratas pengguna muncul. Layar Anda akan terlihat mirip dengan yang berikut ini.


Test recommender
Recommendation parameters
User ID
This is the USER_ID you want to get personalized re-ranked item recommendations for. This USER_ID needs to be present in your user-interactions or user dataset.
1234
Filter name- optional
Choose an existing filter to apply to your recommendations or create a new filter.
None
View
Create new filter
Promotion - optional Info
Define additional business rules to promote a subset of items in recommendations. The promotion filter you specify applies to these items instead of any filter you specify above.
Cancel
Get recommendations
Recommendations (25)
Up to 25 recommendations are displayed. If you applied a promotion, promoted items are distributed randomly.
Recommendation ID
RID-4d12cd84-7d83-4dd9-b849-158b3e8f9ab8
Item ID
592
380
2571
590
150
296
318
780

Memulai grup dataset Domain (SDK for Java 2.x)

Tutorial ini menunjukkan cara menggunakan SDK for Java 2.x untuk membuat grup dataset Domain untuk domain VIDEO_ON_DEMAND. Dalam tutorial ini, Anda membuat rekomendasi untuk pilihan Top untuk kasus penggunaan Anda.

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Prasyarat

Berikut ini adalah langkah-langkah prasyarat untuk menyelesaikan tutorial ini:

- Selesaikan Memulai prasyarat untuk mengatur izin yang diperlukan dan membuat data pelatihan. Jika Anda juga menyelesaikanMemulai grup dataset Domain (konsol), Anda dapat menggunakan kembali data sumber yang sama. Jika Anda menggunakan data sumber Anda sendiri, pastikan bahwa data Anda diformat seperti dalam prasyarat.
- Siapkan lingkungan SDK for Java 2.x AWS dan kredensialnya seperti yang ditentukan dalam Menyiapkan prosedur di Panduan AWS SDK for Java 2.x Pengembang.AWS SDK for Java 2.x

Tutorial

Pada langkah-langkah berikut, Anda menyiapkan project untuk menggunakan paket Amazon Personalize dan membuat Amazon Personalize SDK for Java 2.x client. Kemudian Anda mengimpor data, membuat pemberi rekomendasi untuk pilihan Teratas untuk kasus penggunaan Anda, dan mendapatkan rekomendasi.

Langkah 1: Siapkan project Anda untuk menggunakan paket Amazon Personalize

Setelah menyelesaikan prasyarat, tambahkan dependensi Amazon Personalize ke file pom.xml dan impor paket Amazon Personalize.

1. Tambahkan dependensi berikut ke file pom.xml Anda. Nomor versi terbaru mungkin berbeda dari kode contoh.

```
<dependency>
 <groupId>software.amazon.awssdk</groupId>
 <artifactId>personalize</artifactId>
 <version>2.16.83</version>
```

```
</dependency>
<dependency>
 <groupId>software.amazon.awssdk</groupId>
 <artifactId>personalizeruntime</artifactId>
 <version>2.16.83</version>
</dependency>
<dependency>
 <groupId>software.amazon.awssdk</groupId>
 <artifactId>personalizeevents</artifactId>
 <version>2.16.83</version>
</dependency>
```

2. Tambahkan pernyataan impor berikut ke proyek Anda.

```
// import client packages
import software.amazon.awssdk.services.personalize.PersonalizeClient;
import software.amazon.awssdk.services.personalizeruntime.PersonalizeRuntimeClient;
// Amazon Personalize exception package
import software.amazon.awssdk.services.personalize.model.PersonalizeException;
// schema packages
import software.amazon.awssdk.services.personalize.model.CreateSchemaRequest;
// dataset group packages
import software.amazon.awssdk.services.personalize.model.CreateDatasetGroupRequest;
import software.amazon.awssdk.services.personalize.model.DescribeDatasetGroupRequest;
// dataset packages
import software.amazon.awssdk.services.personalize.model.CreateDatasetRequest;
// dataset import job packages
import
 software.amazon.awssdk.services.personalize.model.CreateDatasetImportJobRequest;
import software.amazon.awssdk.services.personalize.model.DataSource;
import software.amazon.awssdk.services.personalize.model.DatasetImportJob;
import
 software.amazon.awssdk.services.personalize.model.DescribeDatasetImportJobRequest;
// recommender packages
import software.amazon.awssdk.services.personalize.model.CreateRecommenderRequest;
import software.amazon.awssdk.services.personalize.model.CreateRecommenderResponse;
import software.amazon.awssdk.services.personalize.model.DescribeRecommenderRequest;
// get recommendations packages
import
 software.amazon.awssdk.services.personalizeruntime.model.GetRecommendationsRequest;
import
 software.amazon.awssdk.services.personalizeruntime.model.GetRecommendationsResponse;
import software.amazon.awssdk.services.personalizeruntime.model.PredictedItem;
// Java time utility package
```

```
import java.time.Instant;
```

Langkah 2: Buat Amazon Personalisasi klien

Setelah menambahkan dependensi Amazon Personalize ke file pom.xml dan mengimpor paket yang diperlukan, buat klien Amazon Personalize berikut:

```
PersonalizeClient personalizeClient = PersonalizeClient.builder()
  .region(region)
  .build();

PersonalizeRuntimeClient personalizeRuntimeClient = PersonalizeRuntimeClient.builder()
  .region(region)
  .build();
```

Langkah 3: Impor data

Setelah Anda menginisialisasi klien Amazon Personalize, impor data historis yang Anda buat saat Anda menyelesaikan. Memulai prasyarat Untuk mengimpor data historis ke Amazon Personalize, lakukan hal berikut:

1. Simpan skema Avro berikut sebagai file JSON di direktori kerja Anda. Skema ini cocok dengan kolom dalam file CSV yang Anda buat saat Anda menyelesaikan. Membuat data pelatihan (Grup dataset domain)

```
{
  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "ITEM_ID",
          "type": "string"
      },
      {
          "name": "EVENT_TYPE",
          "type": "string"
```

```
      },
      {
          "name": "TIMESTAMP",
          "type": "long"
      }
  ],
  "version": "1.0"
}
```

2. Gunakan `createDomainSchema` metode berikut untuk membuat skema domain di Amazon Personalize. Teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalize, nama untuk skema Anda, `VIDEO_ON_DEMAND` untuk domain, dan jalur file untuk file JSON skema yang Anda buat pada langkah sebelumnya. Metode ini mengembalikan Amazon Resource Name (ARN) skema baru Anda. Simpan untuk digunakan nanti.

```
    public static String createDomainSchema(PersonalizeClient personalizeClient,
String schemaName, String domain,
            String filePath) {

        String schema = null;
        try {
            schema = new String(Files.readAllBytes(Paths.get(filePath)));
        } catch (IOException e) {
            System.out.println(e.getMessage());
        }

        try {
            CreateSchemaRequest createSchemaRequest = CreateSchemaRequest.builder()
                    .name(schemaName)
                    .domain(domain)
                    .schema(schema)
                    .build();

            String schemaArn =
personalizeClient.createSchema(createSchemaRequest).schemaArn();

            System.out.println("Schema arn: " + schemaArn);

            return schemaArn;

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
```

```
        }
        return "";
    }
```

3. Buat grup dataset. Gunakan `createDomainDatasetGroup` metode berikut untuk membuat grup dataset domain. Teruskan parameter berikut ini sebagai parameter: klien layanan Amazon Personalize, nama untuk grup kumpulan data, dan teruskan `VIDEO_ON_DEMAND` untuk domain. Metode mengembalikan ARN dari grup dataset baru Anda. Simpan untuk digunakan nanti.

```
    public static String createDomainDatasetGroup(PersonalizeClient
personalizeClient,
            String datasetGroupName,
            String domain) {

        try {
            CreateDatasetGroupRequest createDatasetGroupRequest =
CreateDatasetGroupRequest.builder()
                    .name(datasetGroupName)
                    .domain(domain)
                    .build();
            return
personalizeClient.createDatasetGroup(createDatasetGroupRequest).datasetGroupArn();
        } catch (PersonalizeException e) {
            System.out.println(e.awsErrorDetails().errorMessage());
        }
        return "";
    }
```

4. Buat kumpulan data interaksi Item. Gunakan `createDataset` metode berikut untuk membuat kumpulan data interaksi Item. Teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalize, nama untuk kumpulan data Anda, ARN skema Anda, ARN grup kumpulan data Anda, dan untuk jenis kumpulan data. `Interactions` Metode mengembalikan ARN dari dataset baru Anda. Simpan untuk digunakan nanti.

```
    public static String createDataset(PersonalizeClient personalizeClient,
            String datasetName,
            String datasetGroupArn,
            String datasetType,
            String schemaArn) {
        try {
            CreateDatasetRequest request = CreateDatasetRequest.builder()
                    .name(datasetName)
```

```
                    .datasetGroupArn(datasetGroupArn)
                    .datasetType(datasetType)
                    .schemaArn(schemaArn)
                    .build();

            String datasetArn = personalizeClient.createDataset(request)
                    .datasetArn();
            System.out.println("Dataset " + datasetName + " created.");
            return datasetArn;

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

5. Impor data Anda dengan pekerjaan impor dataset. Gunakan `createPersonalizeDatasetImportJob` metode berikut untuk membuat pekerjaan impor dataset.

   Teruskan parameter berikut ini sebagai parameter: klien layanan Amazon Personalize, nama untuk pekerjaan, dan ARN kumpulan data Interaksi Anda. Lewati jalur bucket Amazon S3 (`s3://`*`bucket name`*`/`*`folder name`*`/ratings.csv`) tempat Anda menyimpan data pelatihan, dan ARN peran layanan Anda. Anda menciptakan peran ini sebagai bagian dariMemulai prasyarat. Metode mengembalikan ARN dari pekerjaan impor dataset Anda. Secara opsional simpan untuk digunakan nanti.

```
    public static String createPersonalizeDatasetImportJob(PersonalizeClient
personalizeClient,
            String jobName,
            String datasetArn,
            String s3BucketPath,
            String roleArn) {

        long waitInMilliseconds = 60 * 1000;
        String status;
        String datasetImportJobArn;

        try {
            DataSource importDataSource = DataSource.builder()
                    .dataLocation(s3BucketPath)
```

```
                    .build();

            CreateDatasetImportJobRequest createDatasetImportJobRequest =
CreateDatasetImportJobRequest.builder()
                    .datasetArn(datasetArn)
                    .dataSource(importDataSource)
                    .jobName(jobName)
                    .roleArn(roleArn)
                    .build();

            datasetImportJobArn =
personalizeClient.createDatasetImportJob(createDatasetImportJobRequest)
                    .datasetImportJobArn();
            DescribeDatasetImportJobRequest describeDatasetImportJobRequest =
DescribeDatasetImportJobRequest.builder()
                    .datasetImportJobArn(datasetImportJobArn)
                    .build();

            long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            while (Instant.now().getEpochSecond() < maxTime) {

                DatasetImportJob datasetImportJob = personalizeClient
                        .describeDatasetImportJob(describeDatasetImportJobRequest)
                        .datasetImportJob();

                status = datasetImportJob.status();
                System.out.println("Dataset import job status: " + status);

                if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }
            return datasetImportJobArn;

        } catch (PersonalizeException e) {
            System.out.println(e.awsErrorDetails().errorMessage());
        }
        return "";
```

```
    }
```

Langkah 4: Buat pemberi rekomendasi

Setelah pekerjaan impor dataset Anda selesai, Anda siap membuat pemberi rekomendasi. Gunakan `createRecommender` metode berikut untuk membuat pemberi rekomendasi. Teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalize, nama untuk pemberi rekomendasi, Nama Sumber Daya Amazon (ARN) grup dataset Anda, dan untuk resep ARN. `arn:aws:personalize:::recipe/aws-vod-top-picks` Metode ini mengembalikan ARN dari pemberi rekomendasi baru Anda. Simpan untuk digunakan nanti.

```
   public static String createRecommender(PersonalizeClient personalizeClient,
            String name,
            String datasetGroupArn,
            String recipeArn) {

        long maxTime = 0;
        long waitInMilliseconds = 30 * 1000; // 30 seconds
        String recommenderStatus = "";

        try {
            CreateRecommenderRequest createRecommenderRequest =
CreateRecommenderRequest.builder()
                    .datasetGroupArn(datasetGroupArn)
                    .name(name)
                    .recipeArn(recipeArn)
                    .build();

            CreateRecommenderResponse recommenderResponse = personalizeClient
                    .createRecommender(createRecommenderRequest);
            String recommenderArn = recommenderResponse.recommenderArn();
            System.out.println("The recommender ARN is " + recommenderArn);

            DescribeRecommenderRequest describeRecommenderRequest =
DescribeRecommenderRequest.builder()
                    .recommenderArn(recommenderArn)
                    .build();

            maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            while (Instant.now().getEpochSecond() < maxTime) {
```

```
                recommenderStatus =
personalizeClient.describeRecommender(describeRecommenderRequest).recommender()
                        .status();
                System.out.println("Recommender status: " + recommenderStatus);

                if (recommenderStatus.equals("ACTIVE") ||
recommenderStatus.equals("CREATE FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }
            return recommenderArn;

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

Langkah 5: Dapatkan rekomendasi

Setelah Anda membuat pemberi rekomendasi, Anda menggunakannya untuk mendapatkan rekomendasi. Gunakan `getRecs` metode berikut untuk mendapatkan rekomendasi bagi pengguna. Teruskan sebagai parameter klien runtime Amazon Personalize, Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi yang Anda buat pada langkah sebelumnya, dan ID pengguna (misalnya,). 123 Metode ini mencetak daftar item yang direkomendasikan ke layar.

```
    public static void getRecs(PersonalizeRuntimeClient personalizeRuntimeClient,
String recommenderArn,
            String userId) {

        try {
            GetRecommendationsRequest recommendationsRequest =
GetRecommendationsRequest.builder()
                    .recommenderArn(recommenderArn)
                    .numResults(20)
                    .userId(userId)
                    .build();
```

```
            GetRecommendationsResponse recommendationsResponse =
personalizeRuntimeClient
                    .getRecommendations(recommendationsRequest);
            List<PredictedItem> items = recommendationsResponse.itemList();

            for (PredictedItem item : items) {
                System.out.println("Item Id is : " + item.itemId());
                System.out.println("Item score is : " + item.score());
            }
        } catch (AwsServiceException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

Memulai grup dataset Domain (SDK for Python (Boto3))

Tutorial ini menunjukkan cara menggunakan SDK for Python (Boto3) untuk membuat grup dataset Domain untuk domain VIDEO_ON_DEMAND. Dalam tutorial ini, Anda membuat rekomendasi untuk pilihan Top untuk kasus penggunaan Anda.

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Topik

- Prasyarat
- Tutorial
- Memulai menggunakan Amazon Personalize APIs dengan notebook Jupyter (IPython)

Prasyarat

Berikut ini adalah langkah-langkah prasyarat untuk menggunakan contoh Python dalam panduan ini:

- Selesaikan Memulai prasyarat untuk mengatur izin yang diperlukan dan membuat data pelatihan. Jika Anda menggunakan data sumber Anda sendiri, pastikan bahwa data Anda diformat seperti dalam prasyarat.

- Siapkan AWS SDK untuk Python (Boto3) lingkungan Anda seperti yang ditentukan dalamMenyiapkan AWS SDKs.

Tutorial

Dalam langkah-langkah berikut, Anda memverifikasi lingkungan Anda dan membuat SDK for Python (Boto3) klien untuk Amazon Personalize. Kemudian Anda mengimpor data, membuat pemberi rekomendasi untuk pilihan Teratas untuk kasus penggunaan Anda, dan mendapatkan rekomendasi.

Langkah 1: Verifikasi lingkungan Python Anda dan buat klien boto3

Setelah Anda menyelesaikan prasyarat, jalankan contoh Python berikut untuk mengonfirmasi bahwa lingkungan Anda dikonfigurasi dengan benar. Kode ini juga menciptakan Amazon Personalize boto3 klien yang Anda gunakan dalam tutorial ini. Jika lingkungan Anda dikonfigurasi dengan benar, daftar resep yang tersedia ditampilkan dan Anda dapat menjalankan contoh lain dalam tutorial ini.

```
import boto3

personalizeRt = boto3.client('personalize-runtime')
personalize = boto3.client('personalize')

response = personalize.list_recipes()

for recipe in response['recipes']:
    print (recipe)
```

Langkah 2: Impor data

Setelah Anda membuat Amazon Personalisasi klien boto3 dan memverifikasi lingkungan Anda, impor data historis yang Anda buat saat Anda menyelesaikan. Memulai prasyarat Untuk mengimpor data historis ke Amazon Personalize, lakukan hal berikut:

1. Gunakan kode berikut untuk membuat skema di Amazon Personalize. Ganti `gs-domain-interactions-schema` dengan nama untuk skema.

   ```
   import json
   schema = {
     "type": "record",
     "name": "Interactions",
     "namespace": "com.amazonaws.personalize.schema",
   ```

```
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "ITEM_ID",
          "type": "string"
      },
      {
          "name": "EVENT_TYPE",
          "type": "string"
      },
      {
          "name": "TIMESTAMP",
          "type": "long"
      }
  ],
  "version": "1.0"
}

create_interactions_schema_response = personalize.create_schema(
    name='gs-domain-interactions-schema',
    schema=json.dumps(schema),
    domain='VIDEO_ON_DEMAND'
)

interactions_schema_arn = create_interactions_schema_response['schemaArn']
print(json.dumps(create_interactions_schema_response, indent=2))
```

2. Buat grup dataset dengan kode berikut. Ganti `dataset group name` dengan nama untuk grup dataset.

```
response = personalize.create_dataset_group(
  name = 'dataset group name',
  domain = 'VIDEO_ON_DEMAND'
)
dsg_arn = response['datasetGroupArn']

description = personalize.describe_dataset_group(datasetGroupArn = dsg_arn)
['datasetGroup']

print('Name: ' + description['name'])
```

```
print('ARN: ' + description['datasetGroupArn'])
print('Status: ' + description['status'])
```

3. Buat kumpulan data interaksi Item di grup kumpulan data baru Anda dengan kode berikut. Beri nama kumpulan data dan berikan `schema_arn` dan `dataset_group_arn` dari langkah sebelumnya.

```
response = personalize.create_dataset(
    name = 'interactions-dataset-name',
    schemaArn = interactions_schema_arn,
    datasetGroupArn = dsg_arn,
    datasetType = 'INTERACTIONS'
)

dataset_arn = response['datasetArn']
```

4. Impor data Anda dengan pekerjaan impor dataset dengan kode berikut. Kode menggunakan metode describe_dataset_import_job untuk melacak status pekerjaan.

   Teruskan parameter berikut sebagai parameter: nama untuk pekerjaan, `dataset_arn` dari langkah sebelumnya, jalur bucket Amazon S3 (`s3://`*`bucket name`*`/`*`folder name`*`/ratings.csv`) tempat Anda menyimpan data pelatihan, dan ARN peran layanan IAM Anda. Anda menciptakan peran ini sebagai bagian dariMemulai prasyarat. Amazon Personalize memerlukan izin untuk mengakses bucket. Untuk informasi tentang pemberian akses, lihatMemberikan Amazon Personalisasi akses ke sumber daya Amazon S3.

```
import time
response = personalize.create_dataset_import_job(
    jobName = 'JobName',
    datasetArn = 'dataset_arn',
    dataSource = {'dataLocation':'s3://amzn-s3-demo-bucket/filename.csv'},
    roleArn = 'role_arn'
)

dataset_interactions_import_job_arn = response['datasetImportJobArn']

description = personalize.describe_dataset_import_job(
    datasetImportJobArn = dataset_interactions_import_job_arn)['datasetImportJob']

print('Name: ' + description['jobName'])
print('ARN: ' + description['datasetImportJobArn'])
print('Status: ' + description['status'])
```

```
max_time = time.time() + 3*60*60 # 3 hours
while time.time() < max_time:
    describe_dataset_import_job_response = personalize.describe_dataset_import_job(
        datasetImportJobArn = dataset_interactions_import_job_arn
    )
    status = describe_dataset_import_job_response["datasetImportJob"]['status']
    print("Interactions DatasetImportJob: {}".format(status))

    if status == "ACTIVE" or status == "CREATE FAILED":
        break

    time.sleep(60)
```

Langkah 4: Buat pemberi rekomendasi

Setelah pekerjaan impor dataset Anda selesai, Anda siap membuat pemberi rekomendasi. Gunakan kode berikut untuk membuat pemberi rekomendasi. Berikan yang berikut ini sebagai parameter: nama untuk pemberi rekomendasi, Nama Sumber Daya Amazon (ARN) grup dataset Anda, dan `arn:aws:personalize:::recipe/aws-vod-top-picks` untuk resep ARN. Kode menggunakan metode describe_recommended untuk melacak status pemberi rekomendasi.

```
import time
create_recommender_response = personalize.create_recommender(
  name = 'gs-python-top-picks',
  recipeArn = 'arn:aws:personalize:::recipe/aws-vod-top-picks',
  datasetGroupArn = dsg_arn
)
recommender_arn = create_recommender_response['recommenderArn']

print('Recommender ARN:' + recommender_arn)
max_time = time.time() + 3*60*60 # 3 hours
while time.time() < max_time:

    version_response = personalize.describe_recommender(
        recommenderArn = recommender_arn
    )
    status = version_response["recommender"]["status"]

    if status == "ACTIVE":
        print("Creation succeeded for {}".format(recommender_arn))
```

```
    elif status == "CREATE FAILED":
        print("Creation failed for {}".format(recommender_arn))

    if status == "ACTIVE":
        break
    else:
        print("Recommender creation is still in progress")

    time.sleep(60)
```

Langkah 5: Dapatkan rekomendasi

Setelah Anda membuat pemberi rekomendasi, Anda menggunakannya untuk mendapatkan rekomendasi dengan kode berikut. Teruskan sebagai parameter Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi yang Anda buat pada langkah sebelumnya, dan ID pengguna (misalnya,). 123 Metode ini mencetak daftar item yang direkomendasikan.

```
response = personalizeRt.get_recommendations(
    recommenderArn = "arn:aws:personalize:us-west-2:014025156336:recommender/gs-python-
top-picks-89",
    userId = '123'
)
print("Recommended items")
for item in response['itemList']:
    print (item['itemId'])
```

Memulai menggunakan Amazon Personalize APIs dengan notebook Jupyter (IPython)

Untuk mulai membuat grup kumpulan data Domain dengan notebook Jupyter, kloning atau unduh serangkaian buku catatan yang ditemukan di folder notebooks_managed_domains dari repositori Amazon Personalize samples. Notebook memandu Anda mengimpor data pelatihan, membuat pemberi rekomendasi, dan mendapatkan rekomendasi dengan Amazon Personalize.

> **Note**
>
> Sebelum memulai dengan notebook, pastikan untuk membangun lingkungan Anda mengikuti langkah-langkah di README.md

Memulai grup dataset Domain (SDK untuk JavaScript v3)

Tutorial ini menunjukkan cara menggunakan AWS SDK untuk JavaScript v3 untuk membuat grup dataset Domain untuk domain VIDEO_ON_DEMAND. Dalam tutorial ini, Anda membuat rekomendasi untuk pilihan Top untuk kasus penggunaan Anda.

Untuk melihat kode yang digunakan dalam tutorial ini GitHub, lihat contoh kode Amazon Personalize untuk SDK for JavaScript v3 di repositori Contoh Kode AWS SDK.

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Topik

- Prasyarat
- Tutorial

Prasyarat

Berikut ini adalah langkah-langkah prasyarat untuk menyelesaikan tutorial ini:

- Selesaikan Memulai prasyarat untuk mengatur izin yang diperlukan dan membuat data pelatihan. Jika Anda juga menyelesaikanMemulai grup dataset Domain (konsol), Anda dapat menggunakan kembali data sumber yang sama. Jika Anda menggunakan data sumber Anda sendiri, pastikan bahwa data Anda diformat seperti dalam prasyarat.
- Siapkan SDK untuk JavaScript dan AWS kredensialnya seperti yang ditentukan dalam Menyiapkan SDK untuk JavaScript prosedur di Panduan Pengembang.AWS SDK untuk JavaScript

Tutorial

Pada langkah-langkah berikut, Anda menginstal dependensi yang diperlukan. Kemudian Anda membuat grup kumpulan data, mengimpor data, membuat pemberi rekomendasi untuk pilihan Teratas untuk kasus penggunaan Anda, dan mendapatkan rekomendasi.

Jika Anda menggunakan Node.js, Anda dapat menjalankan setiap contoh kode dengan menyimpan sampel sebagai JavaScript file dan kemudian menjalankannya`node <fileName.js>`.

Langkah 1: Instal dependensi Amazon Personalisasi

Setelah Anda menyelesaikan prasyarat, instal dependensi Amazon Personalize berikut:

- @aws -sdk/klien-personalisasi
- @aws -sdk/ client-personalize-runtime
- @aws -sdk/ client-personalize-events (opsional untuk tutorial ini, tetapi diperlukan jika Anda ingin merekam acara setelah Anda membuat pemberi rekomendasi)

Berikut ini adalah contoh `package.json` file yang dapat Anda gunakan. Untuk menginstal dependensi dengan Node.js, navigasikan ke tempat Anda menyimpan `package.json` file dan menjalankannya. `npm install`

```
{
  "name": "personalize-js-project",
  "version": "1.0.0",
  "description": "personalize operations",
  "type": "module",
  "author": "Author Name <email@address.com>",
  "license": "ISC",
  "dependencies": {
    "@aws-sdk/client-personalize": "^3.350.0",
    "@aws-sdk/client-personalize-events": "^3.350.0",
    "@aws-sdk/client-personalize-runtime": "^3.350.0",
    "fs": "^0.0.1-security"
  },
  "compilerOptions": {
    "resolveJsonModule": true,
    "esModuleInterop": true
  }
}
```

Langkah 2: Buat Amazon Personalisasi klien

Setelah Anda menginstal dependensi, buat klien Amazon Personalize Anda. Dalam tutorial ini, contoh kode mengasumsikan Anda membuat klien dalam file bernama `personalizeClients.js` disimpan dalam direktori bernama`libs`.

Berikut ini adalah contoh `personalizeClient.js` file.

```
import { PersonalizeClient } from "@aws-sdk/client-personalize";
```

```
import { PersonalizeRuntimeClient } from "@aws-sdk/client-personalize-runtime";
import { PersonalizeEventsClient } from "@aws-sdk/client-personalize-events";
// Set your AWS region.
const REGION = "region"; //e.g. "us-east-1"

const personalizeClient = new PersonalizeClient({ region: REGION});
const personalizeEventsClient = new PersonalizeEventsClient({ region: REGION});
const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region: REGION});

export { personalizeClient, personalizeEventsClient, personalizeRuntimeClient };
```

Langkah 3: Impor data

Setelah Anda membuat klien Amazon Personalize, impor data historis yang Anda buat saat Anda menyelesaikan. Memulai prasyarat Untuk mengimpor data historis ke Amazon Personalize, lakukan hal berikut:

1. Simpan skema Avro berikut sebagai file JSON di direktori kerja Anda. Skema ini cocok dengan kolom dalam file CSV yang Anda buat saat Anda menyelesaikan. Membuat data pelatihan (Grup dataset domain)

   ```
   {
     "type": "record",
     "name": "Interactions",
     "namespace": "com.amazonaws.personalize.schema",
     "fields": [
         {
             "name": "USER_ID",
             "type": "string"
         },
         {
             "name": "ITEM_ID",
             "type": "string"
         },
         {
             "name": "EVENT_TYPE",
             "type": "string"
         },
         {
             "name": "TIMESTAMP",
             "type": "long"
         }
     ],
   ```

```
  "version": "1.0"
}
```

2. Buat skema domain di Amazon Personalize dengan kode berikut. `createDomainSchema.js` Ganti SCHEMA_PATH dengan path ke file schema.json yang baru saja Anda buat. Perbarui `createSchemaParam` untuk menentukan nama untuk skema, dan untuk `domain` menentukan`VIDEO_ON_DEMAND`.

```
// Get service clients module and commands using ES6 syntax.
import { CreateSchemaCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

import fs from "node:fs";

const schemaFilePath = "SCHEMA_PATH";
let mySchema = "";

try {
  mySchema = fs.readFileSync(schemaFilePath).toString();
} catch (err) {
  mySchema = "TEST"; // for unit tests.
}

// Set the domain schema parameters.
export const createDomainSchemaParam = {
  name: "NAME" /* required */,
  schema: mySchema /* required */,
  domain:
    "DOMAIN" /* required for a domain dataset group, specify ECOMMERCE or
 VIDEO_ON_DEMAND */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSchemaCommand(createDomainSchemaParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
```

```
    console.log("Error", err);
  }
};
run();
```

3. Buat grup kumpulan data domain di Amazon Personalisasi dengan kode berikut. `createDomainDatasetGroup.js` Perbarui `domainDatasetGroupParams` untuk menentukan nama untuk grup kumpulan data, dan untuk `domain` menentukan`VIDEO_ON_DEMAND`.

```
// Get service clients module and commands using ES6 syntax.
import { CreateDatasetGroupCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the domain dataset group parameters.
export const domainDatasetGroupParams = {
  name: "NAME" /* required */,
  domain:
    "DOMAIN" /* required for a domain dsg, specify ECOMMERCE or VIDEO_ON_DEMAND */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetGroupCommand(domainDatasetGroupParams),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

4. Buat kumpulan data interaksi Item di Amazon Personalisasi dengan kode berikut. `createDataset.js` Perbarui `createDatasetParam` untuk menentukan Nama Sumber Daya Amazon (ARN) dari grup kumpulan data dan skema yang baru saja Anda buat, beri nama kumpulan data, dan untuk, tentukan. `datasetType` `Interactions`

```
// Get service clients module and commands using ES6 syntax.
```

```
import { CreateDatasetCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the dataset's parameters.
export const createDatasetParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  datasetType: "DATASET_TYPE" /* required */,
  name: "NAME" /* required */,
  schemaArn: "SCHEMA_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetCommand(createDatasetParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

5. Impor data Anda dengan `createDatasetImportJob.js` kode berikut. Perbarui `datasetImportJobParam` untuk menentukan yang berikut:

   - Tentukan nama untuk pekerjaan tersebut dan tentukan ARN kumpulan data Interaksi Anda.
   - Untuk`dataLocation`, tentukan jalur bucket Amazon S3 (`s3://`*`https://amzn-s3-demo-bucket.s3.region-code.amazonaws.com`*`/`*`folder name`*`/ratings.csv`) tempat Anda menyimpan data pelatihan.
   - Untuk `roleArn` menentukan Nama Sumber Daya Amazon untuk peran layanan Amazon Personalisasi Anda. Anda menciptakan peran ini sebagai bagian dariMemulai prasyarat.

```
// Get service clients module and commands using ES6 syntax.
import { CreateDatasetImportJobCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
```

```
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the dataset import job parameters.
export const datasetImportJobParam = {
  datasetArn: "DATASET_ARN" /* required */,
  dataSource: {
    /* required */
    dataLocation: "S3_PATH",
  },
  jobName: "NAME" /* required */,
  roleArn: "ROLE_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetImportJobCommand(datasetImportJobParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Langkah 4: Buat pemberi rekomendasi

Setelah pekerjaan impor dataset Anda selesai, Anda siap membuat pemberi rekomendasi. Untuk membuat pemberi rekomendasi, gunakan `createRecommender.js` kode berikut. Perbarui `createRecommenderParam` dengan yang berikut ini: Tentukan nama untuk pemberi rekomendasi, tentukan ARN grup kumpulan data Anda, dan untuk tentukan. `recipeArn` `arn:aws:personalize:::recipe/aws-vod-top-picks`

```
// Get service clients module and commands using ES6 syntax.
import { CreateRecommenderCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the recommender's parameters.
```

```
export const createRecommenderParam = {
  name: "NAME" /* required */,
  recipeArn: "RECIPE_ARN" /* required */,
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateRecommenderCommand(createRecommenderParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Langkah 5: Dapatkan rekomendasi

Setelah Anda membuat pemberi rekomendasi, Anda menggunakannya untuk mendapatkan rekomendasi. Gunakan `getRecommendations.js` kode berikut untuk mendapatkan rekomendasi bagi pengguna. Perbarui `getRecommendationsParam` untuk menentukan ARN dari pemberi rekomendasi yang Anda buat pada langkah sebelumnya, dan tentukan ID pengguna (misalnya,). `123`

```
// Get service clients module and commands using ES6 syntax.
import { GetRecommendationsCommand } from "@aws-sdk/client-personalize-runtime";
import { personalizeRuntimeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region: "REGION"});

// Set the recommendation request parameters.
export const getRecommendationsParam = {
  recommenderArn: "RECOMMENDER_ARN" /* required */,
  userId: "USER_ID" /* required */,
  numResults: 15 /* optional */,
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(
      new GetRecommendationsCommand(getRecommendationsParam),
```

```
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Memulai grup Dataset Kustom

> **Important**
>
> Dalam tutorial ini Anda membuat solusi yang menggunakan pelatihan otomatis. Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, pastikan untuk menghapus solusi ketika Anda selesai. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Panduan memulai ini menunjukkan kepada Anda cara memberikan rekomendasi film yang dipersonalisasi untuk pengguna Anda dengan grup kumpulan data khusus dan Resep personalisasi-v2 pengguna resepnya. Tutorial ini menggunakan data historis yang terdiri dari 100.000 peringkat film pada 9.700 film dari 600 pengguna.

Untuk memulai, selesaikan Memulai prasyarat dan kemudian lanjutkan ke salah satuMemulai (konsol),Memulai (AWS CLI),Memulai (SDK for Python (Boto3)), atauMemulai (SDK for Java 2.x).

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Lihat informasi yang lebih lengkap di Persyaratan untuk menghapus sumber daya Amazon Personalize.

Topik

- Memulai (konsol)
- Memulai (AWS CLI)
- Memulai (SDK for Python (Boto3))
- Memulai (SDK for Java 2.x)

Memulai (konsol)

Dalam latihan ini, Anda menggunakan konsol Amazon Personalize untuk membuat grup kumpulan data kustom dengan solusi yang menampilkan rekomendasi film untuk pengguna tertentu. Sebelum Anda memulai latihan ini, tinjau kembaliMemulai prasyarat.

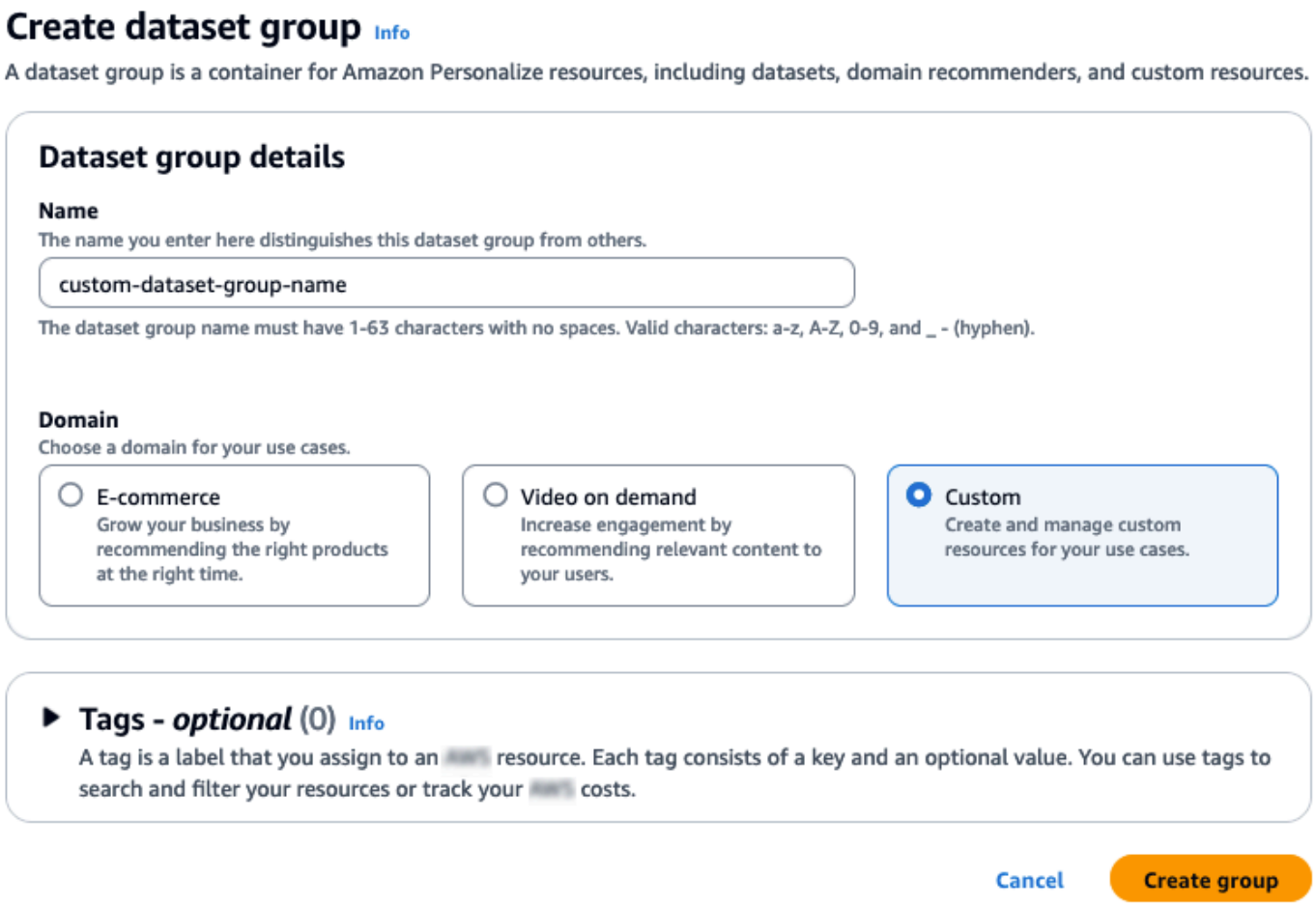
Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Langkah 1: Buat grup dataset dan dataset

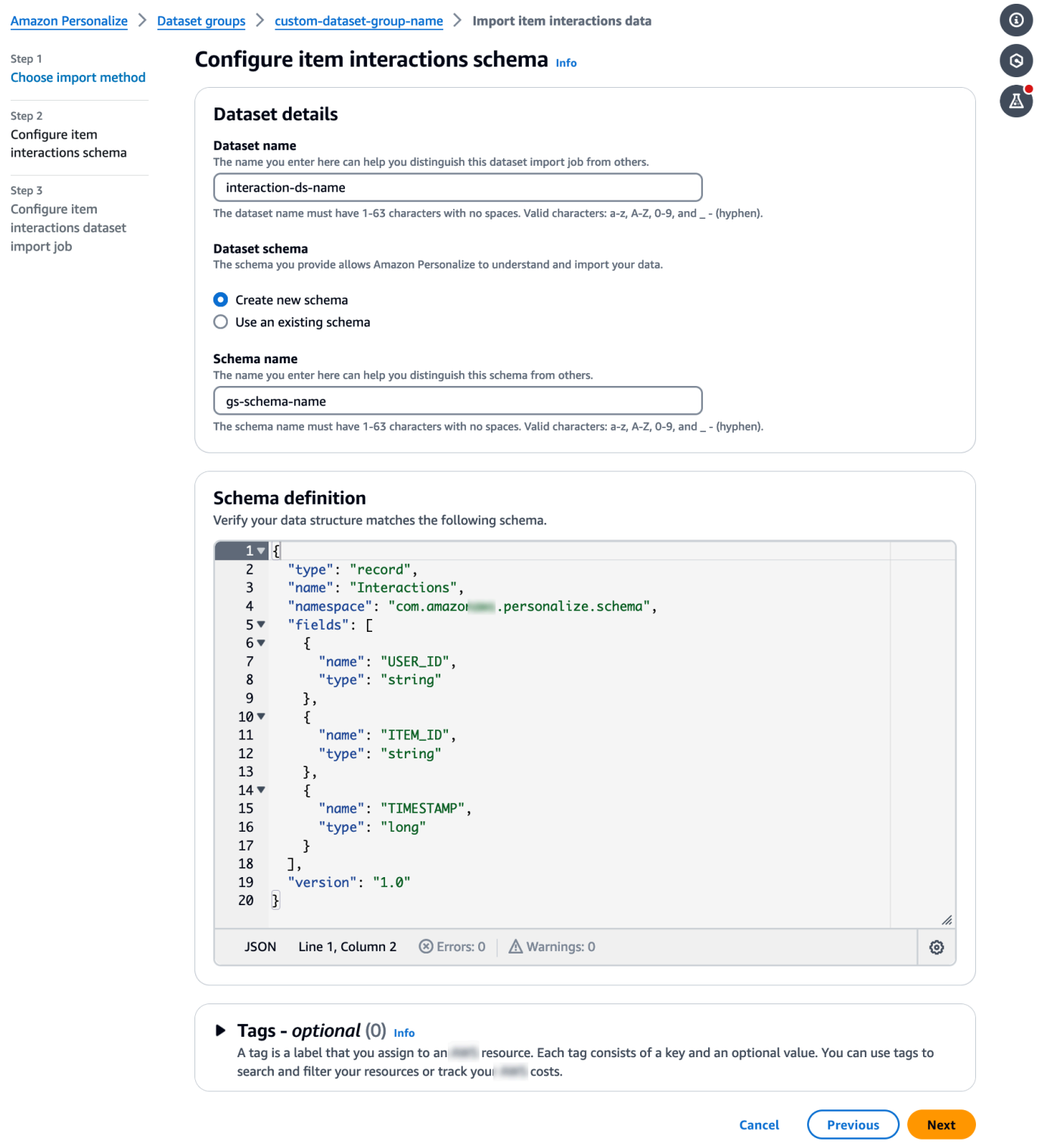
Dalam prosedur ini, Anda pertama kali membuat grup dataset. Selanjutnya, Anda membuat kumpulan data kumpulan data interaksi Item Personalisasi Amazon di grup kumpulan data.

Untuk membuat grup dataset dan dataset

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih Buat grup kumpulan data.
3. Di detail grup Dataset, untuk nama grup Dataset, tentukan nama untuk grup kumpulan data Anda.
4. Untuk Domain pilih Kustom. Layar Anda akan terlihat mirip dengan yang berikut ini:



5. Pilih Buat grup. Halaman Ikhtisar muncul.
6. Pada Langkah 1. Buat kumpulan data dan impor data, pilih Buat kumpulan data dan pilih Set data interaksi item.
7. Pilih Impor data langsung ke Amazon Personalize dataset dan pilih Berikutnya.
8. Di halaman Konfigurasikan kumpulan data interaksi item, untuk nama Dataset, tentukan nama untuk kumpulan data Anda.
9. Untuk skema Dataset, pilih Buat skema baru. Di bagian Definisi skema, skema interaksi Item minimal ditampilkan. Skema cocok dengan header yang sebelumnya Anda tambahkan ke `ratings.csv` file, jadi Anda tidak perlu membuat perubahan apa pun. Jika Anda belum membuat data pelatihan, lihatMemulai prasyarat.
10. Untuk nama Skema, tentukan nama untuk skema baru. Layar Anda akan terlihat mirip dengan yang berikut ini:



11. Pilih Berikutnya. Halaman pekerjaan impor set data Konfigurasikan interaksi item muncul. Selanjutnya, selesaikan Langkah 2: Impor data interaksi item untuk mengimpor data interaksi.

Langkah 2: Impor data interaksi item

Sekarang setelah Anda membuat kumpulan data, saatnya untuk mengimpor data interaksi item ke dalam kumpulan data.

Untuk mengimpor data interaksi item

1. Di halaman pekerjaan impor set data Konfigurasikan interaksi item, untuk Sumber impor data pilih Impor data dari S3.
2. Untuk nama pekerjaan impor Dataset, tentukan nama untuk pekerjaan impor Anda.
3. Di kotak dialog wajib kebijakan bucket S3 tambahan, jika Anda belum memberikan izin Amazon Personalize, ikuti petunjuk untuk menambahkan kebijakan bucket Amazon S3 yang diperlukan.
4. Untuk lokasi Data, tentukan tempat file data film Anda disimpan di Amazon Simple Storage Service (S3). Gunakan sintaks berikut:

   ```
   s3://amzn-s3-demo-bucket/<folder path>/filename.csv
   ```

5. Di bagian Peran IAM, untuk peran layanan IAM, pilih Masukkan peran IAM khusus ARN.
6. Untuk ARN peran IAM Kustom, tentukan peran yang Anda buat. Membuat peran IAM untuk Amazon Personalize

   Detail pekerjaan impor Dataset dan bagian peran IAM harus serupa dengan yang berikut ini:

Configure item interactions dataset import job Info

Dataset import job details

Data import source

- (●) Import data from S3
  Specify the location where your data is stored in S3.
- ( ) Incrementally import data with APIs
  Incrementally import item interactions data with the event ingestion SDK.

Dataset import job name
The name you enter here can help you distinguish this dataset import job from others.

gs-dataset-import-job

The dataset import job name must have 1-63 characters with no spaces. Valid characters: a-z, A-Z, 0-9, and _ - (hyphen).

Data import source

Additional S3 bucket policy required
In addition to the IAM service role defined above, Amazon Personalize also requires you to add a bucket policy to the S3 bucket containing your data files so that it can process them. Follow the instructions described here to add the required bucket policy to your S3 bucket.

Data location Info
Choose the S3 location of your data.

s3://bucket-name/ratings.csv

Your file needs to be in a CSV format and reflect the schema.

IAM Role

IAM service role
Amazon Personalize requires permissions to access your S3 bucket. Choose an existing role with access or create a role in the IAM console with the AmazonPersonalizeFullAccess IAM policy attached.

Enter a custom IAM role ARN ▼

Custom IAM role ARN

arn:aws:iam::accountNumber:role/roleName

After you import data from S3, you can still incrementally import data with the Amazon Personalize console, the AWS Command Line Interface (AWS CLI), or the AWS SDKs.

Publish event metrics to S3 - *optional*

When you create a metric attribution, reports related to this import job can be published to S3 for analysis with your tool of choice.

To track and publish metrics for events, you must create a metric attribution and define event metrics. Create metric attribution

☐ Publish metrics from this import job (you have not created a metric attribution)

7. Biarkan metrik acara Publikasikan ke bagian S3 dan Tag tidak berubah dan pilih Mulai impor. Pekerjaan impor data dimulai dan halaman Ikhtisar ditampilkan. Awalnya, statusnya adalah Create pending (diikuti oleh Create in progress), dan tombol Create solution dinonaktifkan.

   Ketika pekerjaan impor data telah selesai, status berubah menjadi Aktif dan tombol Buat solusi diaktifkan.

8. Sekarang setelah Anda mengimpor data, Anda siap untuk membuat solusiLangkah 3: Buat solusi.

Langkah 3: Buat solusi

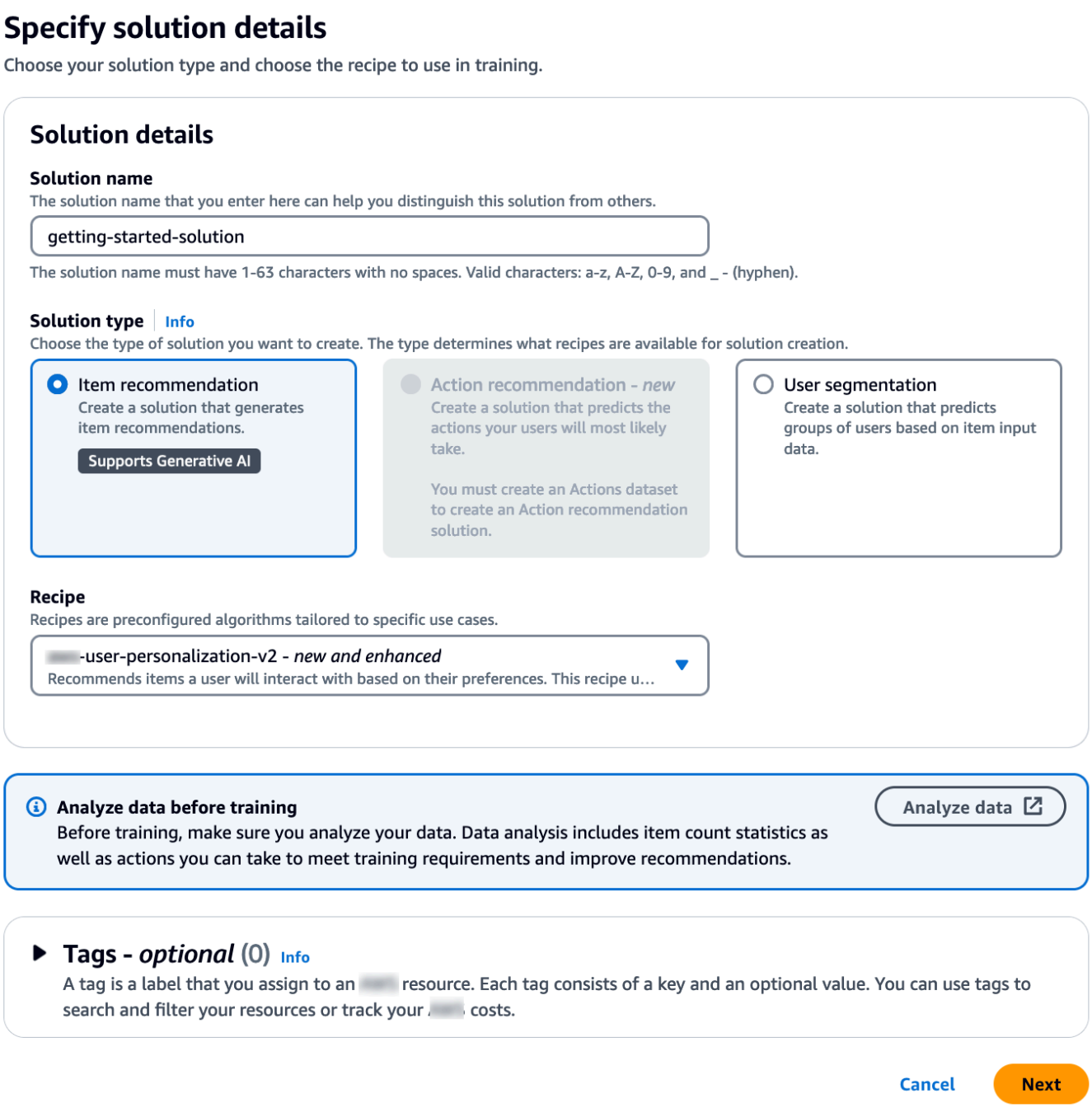
Dalam tutorial ini, Anda menggunakan dataset yang Anda impor Langkah 2: Impor data interaksi item untuk melatih model. Model terlatih disebut sebagai versi solusi.

> **Important**
>
> Dalam tutorial ini Anda membuat solusi yang menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, pastikan untuk menghapus solusi ketika Anda selesai. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Untuk membuat solusi

1. Pada halaman Ikhtisar untuk grup kumpulan data Anda, di Langkah 3. Siapkan sumber daya pelatihan dan rekomendasi pilih Buat solusi.
2. Untuk nama Solusi, tentukan nama untuk solusi Anda.
3. Untuk jenis Solusi pilih Rekomendasi item.
4. Untuk Resep, pilih`aws-user-personalization-v2`.

   Layar Anda akan terlihat mirip dengan yang berikut ini:



5. Pilih Berikutnya. Biarkan bidang konfigurasi Pelatihan tidak berubah. Solusi yang Anda buat secara otomatis melatih model baru setiap 7 hari dan memberi bobot lebih pada data interaksi item terbaru.
6. Pilih Berikutnya dan tinjau detailnya untuk solusinya.
7. Pilih Buat solusi dan halaman detail untuk tampilan solusi. Setelah Anda membuat solusi, Amazon Personalize mulai membuat versi solusi pertama Anda dalam waktu satu jam. Saat

pelatihan dimulai, pelatihan akan muncul di bagian Versi solusi di halaman detail dan Anda dapat memantau statusnya.

Ketika status versi Solusi Aktif, Anda siap untuk pindah keLangkah 4: Buat kampanye.

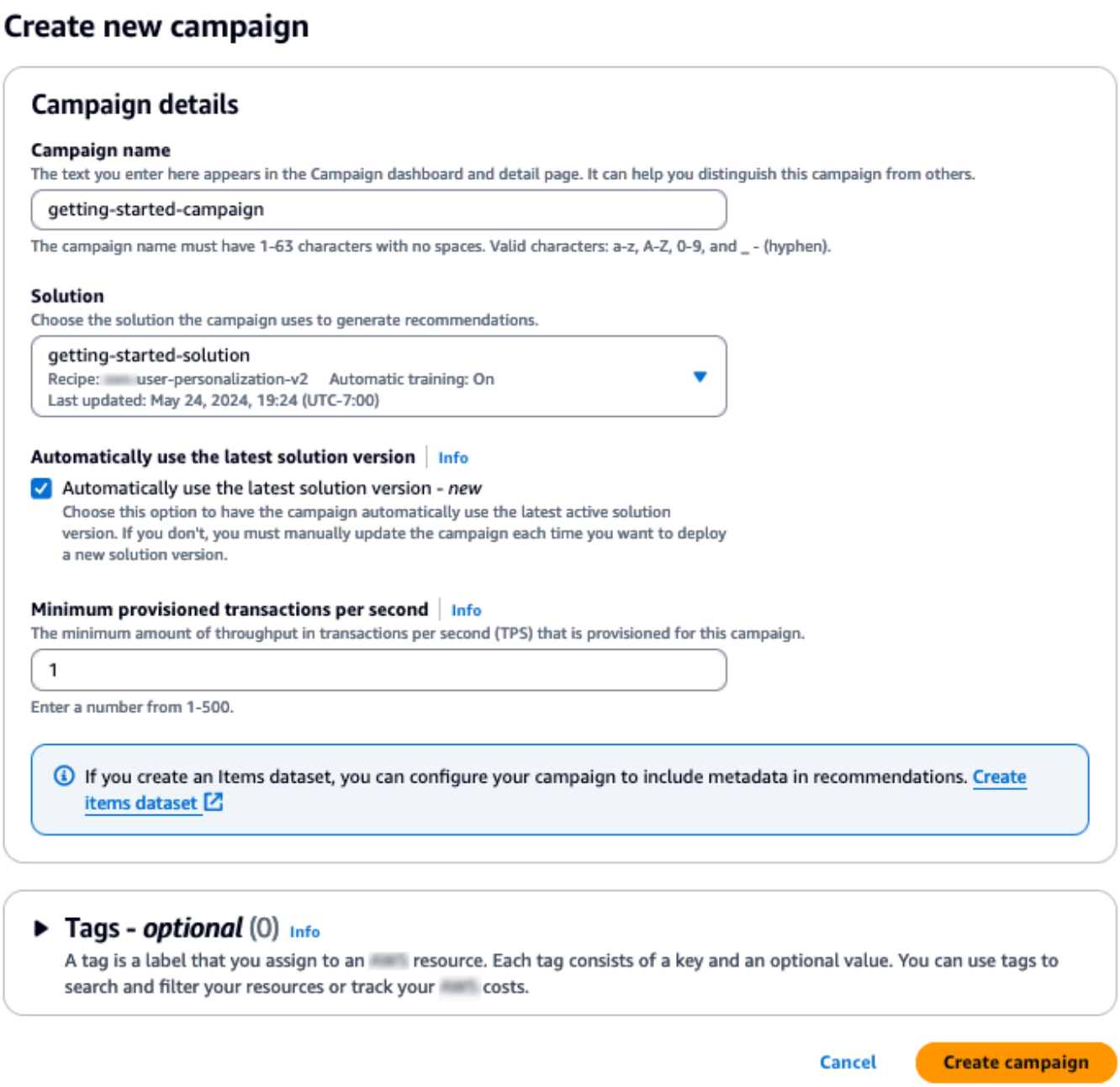
Langkah 4: Buat kampanye

Dalam prosedur ini, Anda membuat kampanye, yang menerapkan versi solusi yang Anda buat di langkah sebelumnya.

Untuk membuat kampanye

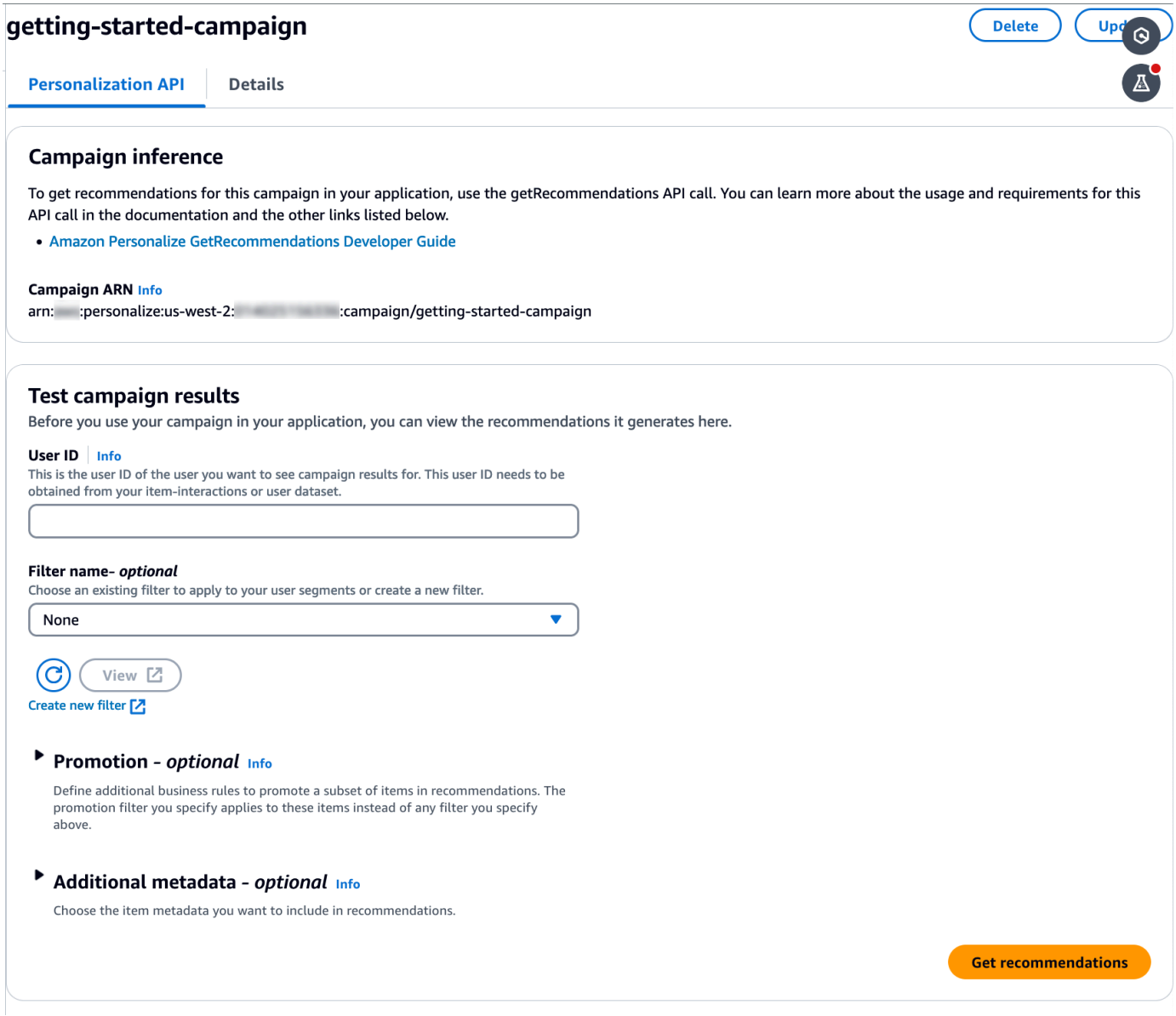
1. Di panel navigasi, perluas Sumber daya kustom dan pilih Kampanye.
2. Pilih Buat kampanye. Halaman Buat kampanye baru akan muncul.
3. Di Detail kampanye, untuk nama Kampanye, tentukan nama untuk kampanye Anda.
4. Untuk Solusi, pilih solusi yang Anda buat di langkah sebelumnya.
5. Pilih Secara otomatis menggunakan versi solusi terbaru. Biarkan semua bidang lainnya tidak berubah.

   Layar Anda akan terlihat mirip dengan yang berikut ini:



6. Pilih Buat kampanye. Pembuatan kampanye dimulai dan halaman detail kampanye dengan bagian API Personalisasi ditampilkan.

   Membuat kampanye dapat memakan waktu beberapa menit. Setelah Amazon Personalize selesai membuat kampanye Anda, halaman diperbarui untuk menampilkan bagian Hasil kampanye uji. Layar Anda akan terlihat mirip dengan yang berikut ini:



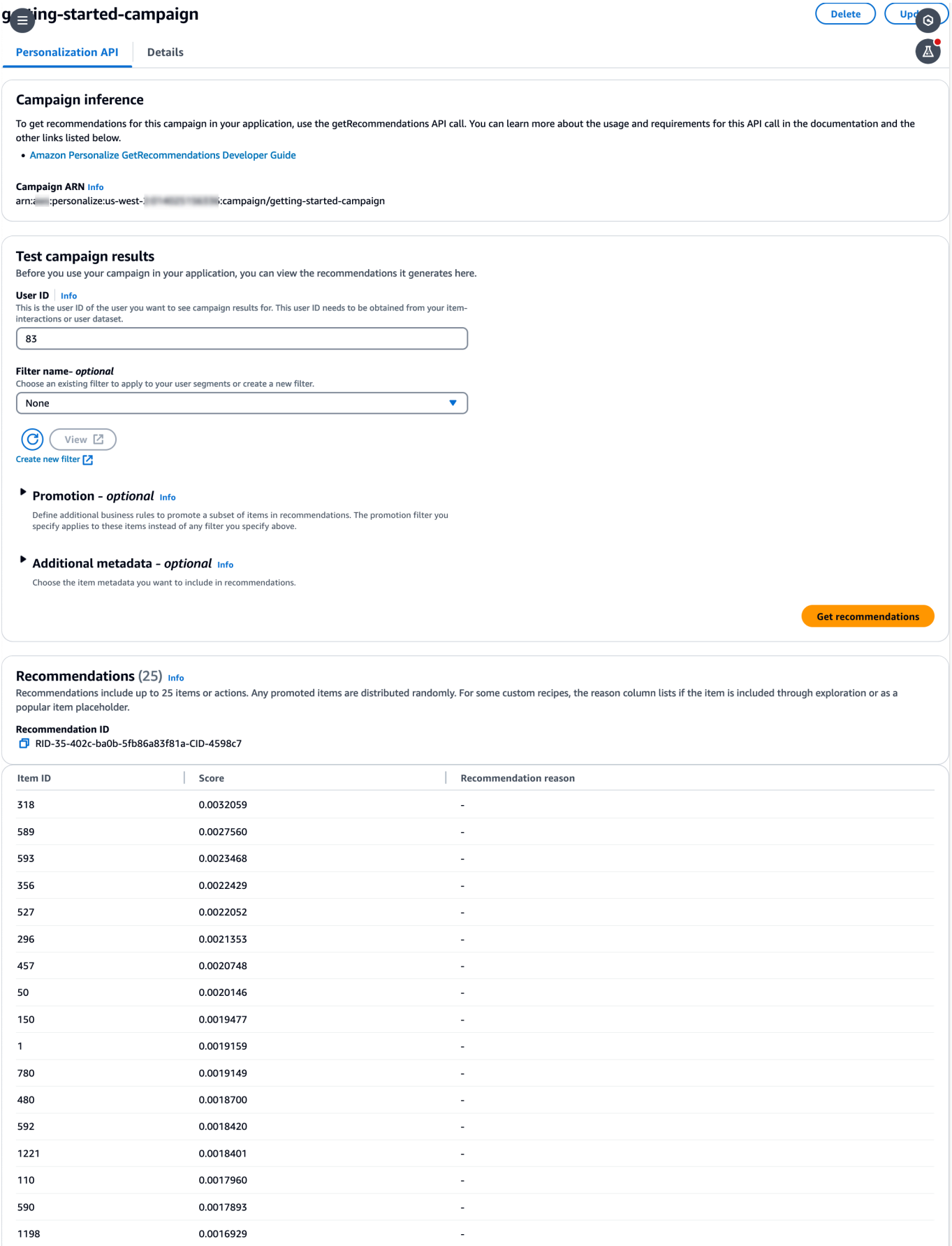
Langkah 5: Dapatkan rekomendasi

Dalam prosedur ini, gunakan kampanye yang Anda buat di langkah sebelumnya untuk mendapatkan rekomendasi.

Untuk mendapatkan rekomendasi

1. Dalam hasil kampanye Uji, untuk ID Pengguna, tentukan nilai dari kumpulan data peringkat, misalnya,**83**. Biarkan semua bidang lainnya tidak berubah.
2. Pilih Dapatkan rekomendasi. Panel Rekomendasi mencantumkan item IDs dan skor untuk item yang direkomendasikan.

Layar Anda akan terlihat mirip dengan yang berikut ini:



getting-started-campaign

Delete Upd

Personalization API | Details

Campaign inference

To get recommendations for this campaign in your application, use the getRecommendations API call. You can learn more about the usage and requirements for this API call in the documentation and the other links listed below.

- Amazon Personalize GetRecommendations Developer Guide

Campaign ARN Info
arn:aws:personalize:us-west-2:[redacted]:campaign/getting-started-campaign

Test campaign results

Before you use your campaign in your application, you can view the recommendations it generates here.

User ID Info
This is the user ID of the user you want to see campaign results for. This user ID needs to be obtained from your item-interactions or user dataset.

83

Filter name- *optional*
Choose an existing filter to apply to your user segments or create a new filter.

None

View

Create new filter

Promotion - *optional* Info

Define additional business rules to promote a subset of items in recommendations. The promotion filter you specify applies to these items instead of any filter you specify above.

Additional metadata - *optional* Info

Choose the item metadata you want to include in recommendations.

Get recommendations

Recommendations (25) Info

Recommendations include up to 25 items or actions. Any promoted items are distributed randomly. For some custom recipes, the reason column lists if the item is included through exploration or as a popular item placeholder.

Recommendation ID
RID-35-402c-ba0b-5fb86a83f81a-CID-4598c7

Item ID	Score	Recommendation reason
318	0.0032059	-
589	0.0027560	-
593	0.0023468	-
356	0.0022429	-
527	0.0022052	-
296	0.0021353	-
457	0.0020748	-
50	0.0020146	-
150	0.0019477	-
1	0.0019159	-
780	0.0019149	-
480	0.0018700	-
592	0.0018420	-
1221	0.0018401	-
110	0.0017960	-
590	0.0017893	-
1198	0.0016929	-
47	0.0016593	-
32	0.0016585	-
364	0.0015704	-
500	0.0015517	-
7361	0.0015233	-

Memulai (AWS CLI)

Dalam latihan ini, Anda menggunakan AWS Command Line Interface (AWS CLI) untuk menjelajahi Amazon Personalize. Anda membuat kampanye yang menampilkan rekomendasi film untuk ID pengguna tertentu.

Sebelum Anda memulai latihan ini, lakukan hal berikut:

- Tinjau MemulaiMemulai prasyarat.
- Mengatur AWS CLI, seperti yang ditentukan dalamMenyiapkan AWS CLI.

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

> **Note**
>
> AWS CLI Perintah dalam latihan ini diuji di Linux. Untuk informasi tentang menggunakan AWS CLI perintah di Windows, lihat Menentukan nilai parameter untuk AWS Command Line Interface dalam Panduan AWS Command Line Interface Pengguna.

Langkah 1: Impor data pelatihan

Ikuti langkah-langkah untuk membuat grup kumpulan data, menambahkan kumpulan data ke grup, lalu mengisi kumpulan data menggunakan data peringkat film.

1. Buat grup dataset dengan menjalankan perintah berikut. Anda dapat mengenkripsi grup kumpulan data dengan meneruskan ARN kunci dan AWS Key Management ServiceARN dari peran IAM yang memiliki izin akses ke kunci tersebut sebagai parameter input. Untuk informasi selengkapnya tentang API, lihat CreateDatasetGroup.

   ```
   aws personalize create-dataset-group --name MovieRatingDatasetGroup --kms-key-
   arn arn:aws:kms:us-west-2:01234567890:key/1682a1e7-a94d-4d92-bbdf-837d3b62315e --
   role-arn arn:aws:iam::01234567890:KMS-key-access
   ```

   Grup kumpulan data ARN ditampilkan, misalnya:

   ```
   {
   ```

```
  "datasetGroupArn": "arn:aws:personalize:us-west-2:acct-id:dataset-group/
MovieRatingDatasetGroup"
}
```

Gunakan `describe-dataset-group` perintah untuk menampilkan grup kumpulan data yang Anda buat, menentukan ARN grup kumpulan data yang dikembalikan.

```
aws personalize describe-dataset-group \
--dataset-group-arn arn:aws:personalize:us-west-2:acct-id:dataset-group/
MovieRatingDatasetGroup
```

Grup dataset dan propertinya ditampilkan, misalnya:

```
{
    "datasetGroup": {
        "name": "MovieRatingDatasetGroup",
        "datasetGroupArn": "arn:aws:personalize:us-west-2:acct-id:dataset-group/
MovieRatingDatasetGroup",
        "status": "ACTIVE",
        "creationDateTime": 1542392161.262,
        "lastUpdatedDateTime": 1542396513.377
    }
}
```

> **Note**
>
> Tunggu hingga grup dataset `status` ditampilkan sebagai AKTIF sebelum membuat kumpulan data dalam grup. Operasi ini biasanya cepat.

Jika Anda tidak ingat grup kumpulan data ARN, gunakan `list-dataset-groups` perintah untuk menampilkan semua grup kumpulan data yang Anda buat, bersama dengan grup tersebut. ARNs

```
aws personalize list-dataset-groups
```

> **Note**
>
> `list-objects`Perintah `describe-object` dan tersedia untuk sebagian besar objek Amazon Personalize. Perintah-perintah ini tidak ditampilkan di sisa latihan ini tetapi mereka tersedia.

2. Buat file skema dalam format JSON dengan menyimpan kode berikut ke file bernama. `MovieRatingSchema.json` Skema cocok dengan header yang sebelumnya Anda tambahkan. `ratings.csv` Nama skema adalah`Interactions`, yang cocok dengan salah satu jenis kumpulan data yang dikenali oleh Amazon Personalize. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

```
{
  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "ITEM_ID",
          "type": "string"
      },
      {
          "name": "TIMESTAMP",
          "type": "long"
      }
  ],
  "version": "1.0"
}
```

3. Buat skema dengan menjalankan perintah berikut. Tentukan file yang Anda simpan di langkah sebelumnya. Contoh menunjukkan file sebagai milik folder saat ini. Untuk informasi selengkapnya tentang API, lihat CreateSchema.

```
aws personalize create-schema \
  --name MovieRatingSchema \
  --schema file://MovieRatingSchema.json
```

Skema Amazon Resource Name (ARN) ditampilkan, misalnya:

```
{
  "schemaArn": "arn:aws:personalize:us-west-2:acct-id:schema/MovieRatingSchema"
}
```

4. Buat dataset kosong dengan menjalankan perintah berikut. Berikan kelompok dataset ARN dan skema ARN yang dikembalikan pada langkah sebelumnya. `dataset-type`Harus cocok dengan skema `name` dari langkah sebelumnya. Untuk informasi selengkapnya tentang API, lihat CreateDataset.

```
aws personalize create-dataset \
  --name MovieRatingDataset \
  --dataset-group-arn arn:aws:personalize:us-west-2:acct-id:dataset-group/
MovieRatingDatasetGroup \
  --dataset-type Interactions \
  --schema-arn arn:aws:personalize:us-west-2:acct-id:schema/MovieRatingSchema
```

Dataset ARN ditampilkan, misalnya:

```
{
  "datasetArn": "arn:aws:personalize:us-west-2:acct-id:dataset/
MovieRatingDatasetGroup/INTERACTIONS"
}
```

5. Tambahkan data pelatihan ke kumpulan data.

   a. Buat pekerjaan impor dataset dengan menjalankan perintah berikut. Berikan nama bucket ARN dan Amazon S3 kumpulan data yang dikembalikan pada langkah sebelumnya. Berikan peran AWS Identity and Access Management (IAM) ARN yang Anda buat. Membuat peran IAM untuk Amazon Personalize Untuk informasi selengkapnya tentang API, lihat CreateDatasetImportJob.

```
aws personalize create-dataset-import-job \
  --job-name MovieRatingImportJob \
  --dataset-arn arn:aws:personalize:us-west-2:acct-id:dataset/
MovieRatingDatasetGroup/INTERACTIONS \
  --data-source dataLocation=s3://amzn-s3-demo-bucket/ratings.csv \
  --role-arn roleArn
```

Pekerjaan impor dataset ARN ditampilkan, misalnya:

```
{
  "datasetImportJobArn": "arn:aws:personalize:us-west-2:acct-id:dataset-import-
job/MovieRatingImportJob"
}
```

b. Periksa status dengan menggunakan `describe-dataset-import-job` perintah. Berikan ARN pekerjaan impor dataset yang dikembalikan pada langkah sebelumnya. Untuk informasi selengkapnya tentang API, lihat DescribeDatasetImportJob.

```
aws personalize describe-dataset-import-job \
  --dataset-import-job-arn arn:aws:personalize:us-west-2:acct-id:dataset-
import-job/MovieRatingImportJob
```

Properti pekerjaan impor dataset, termasuk statusnya, ditampilkan. Awalnya, `status` ditampilkan sebagai CREATE PENDING, misalnya:

```
{
  "datasetImportJob": {
      "jobName": "MovieRatingImportJob",
      "datasetImportJobArn": "arn:aws:personalize:us-west-2:acct-id:dataset-
import-job/MovieRatingImportJob",
      "datasetArn": "arn:aws:personalize:us-west-2:acct-id:dataset/
MovieRatingDatasetGroup/INTERACTIONS",
      "dataSource": {
          "dataLocation": "s3://amzn-s3-demo-bucket/ratings.csv"
      },
      "roleArn": "role-arn",
      "status": "CREATE PENDING",
      "creationDateTime": 1542392161.837,
      "lastUpdatedDateTime": 1542393013.377
  }
}
```

Impor dataset selesai ketika status ditampilkan sebagai AKTIF. Kemudian Anda siap untuk melatih model menggunakan dataset yang ditentukan.

> **Note**
>
> Mengimpor membutuhkan waktu. Tunggu hingga impor dataset selesai sebelum melatih model menggunakan dataset.

Langkah 2: Buat solusi (latih model)

Untuk melatih model, Anda membuat konfigurasi untuk melatih model menggunakan CreateSolution operasi dan membiarkan pelatihan otomatis aktif. Solusi secara otomatis mulai melatih solusi pertama dalam waktu satu jam.

Anda melatih model menggunakan resep dan data pelatihan Anda. Amazon Personalize menyediakan satu set resep yang telah ditentukan. Untuk informasi selengkapnya, lihat Memilih resep. Untuk latihan ini, Anda menggunakan User-Personalization-v 2 resep.

1. Buat konfigurasi untuk melatih model dengan menjalankan perintah berikut. Perintah ini menciptakan solusi yang menggunakan pelatihan otomatis. Secara otomatis membuat versi solusi baru setiap tujuh hari (default).

```
aws personalize create-solution \
  --name MovieSolution \
  --dataset-group-arn arn:aws:personalize:us-west-2:acct-id:dataset-group/
MovieRatingDatasetGroup \
  --recipe-arn arn:aws:personalize:::recipe/aws-user-personalization-v2 \
  --perform-auto-training \
  --solution-config "{\"autoTrainingConfig\": {\"schedulingExpression\": \"rate(7
 days)\"}}"
```

   Solusi ARN ditampilkan, misalnya:

```
{
  "solutionArn": "arn:aws:personalize:us-west-2:acct-id:solution/MovieSolution"
}
```

2. Periksa status buat menggunakan `describe-solution` perintah. Memberikan solusi ARN yang dikembalikan pada langkah sebelumnya. Untuk informasi selengkapnya tentang API, lihat DescribeSolution.

```
aws personalize describe-solution \
  --solution-arn arn:aws:personalize:us-west-2:acct-id:solution/MovieSolution
```

Properti solusi dan pembuatan `status` ditampilkan. Misalnya:

```
{
  "solution": {
      "name": "MovieSolution",
      "solutionArn": "arn:aws:personalize:us-west-2:acct-id:solution/
MovieSolution",
      "performHPO": false,
      "performAutoML": false,
      "recipeArn": "arn:aws:personalize:::recipe/aws-user-personalization-v2",
      "datasetGroupArn": "arn:aws:personalize:us-west-2:acct-id:dataset-group/
MovieRatingDatasetGroup",
      "solutionConfig": {
            "algorithmHyperParameters": {
                "apply_recency_bias": "true"
            },
            "featureTransformationParameters": {},
            "autoTrainingConfig": {
                "schedulingExpression": "rate(7 days)"
            }
       },
      "status": "ACTIVE",
      "creationDateTime": "2021-05-12T16:27:59.819000-07:00",
      "lastUpdatedDateTime": "2021-05-12T16:27:59.819000-07:00"
  }
}
```

3. Dengan pelatihan otomatis, pelatihan versi solusi dimulai dalam satu setelah solusi AKTIF. Setelah pelatihan dimulai, Anda bisa mendapatkan Amazon Resource Name (ARN) versi solusi dengan perintah berikut: ListSolutionVersions

```
aws personalize list-solution-versions --solution-arn arn:aws:personalize:us-
west-2:acct-id:solution/MovieSolution
```

4. Periksa status pelatihan versi solusi dengan menggunakan `describe-solution-version` perintah. Berikan solusi versi ARN yang dikembalikan pada langkah sebelumnya. Untuk informasi selengkapnya tentang API, lihat DescribeSolutionVersion.

```
aws personalize describe-solution-version \
  --solution-version-arn arn:aws:personalize:us-west-2:acct-id:solution/
MovieSolution/version-id
```

Properti versi solusi dan pelatihan `status` ditampilkan. Awalnya, status ditampilkan sebagai CREATE PENDING, misalnya:

```
{
  "solutionVersion": {
      "solutionVersionArn": "arn:aws:personalize:us-west-2:acct-id:solution/
MovieSolution/<version-id>",
      ...,
      "status": "CREATE PENDING"
  }
}
```

5. Ketika versi `status` solusi AKTIF, pelatihan selesai.

   Sekarang Anda dapat meninjau metrik pelatihan dan membuat kampanye menggunakan versi solusi.

   > **Note**
   >
   > Pelatihan membutuhkan waktu. Tunggu hingga pelatihan selesai (status pelatihan versi solusi ditampilkan sebagai AKTIF) sebelum menggunakan versi solusi ini dalam kampanye.

6. Anda dapat memvalidasi kinerja versi solusi dengan meninjau metriknya. Dapatkan metrik untuk versi solusi dengan menjalankan perintah berikut. Berikan solusi versi ARN yang dikembalikan sebelumnya. Untuk informasi selengkapnya tentang API, lihat GetSolutionMetrics.

   ```
   aws personalize get-solution-metrics \
     --solution-version-arn arn:aws:personalize:us-west-2:acct-id:solution/
   MovieSolution/version-id
   ```

   Respons sampel ditampilkan:

   ```
   {
   ```

```
  "solutionVersionArn": "arn:aws:personalize:us-west-2:acct-id:solution/www-
solution/<version-id>",
  "metrics": {
        "coverage": 0.0485,
        "mean_reciprocal_rank_at_25": 0.0381,
        "normalized_discounted_cumulative_gain_at_10": 0.0363,
        "normalized_discounted_cumulative_gain_at_25": 0.0984,
        "normalized_discounted_cumulative_gain_at_5": 0.0175,
        "precision_at_10": 0.0107,
        "precision_at_25": 0.0207,
        "precision_at_5": 0.0107
    }
}
```

Langkah 3: Buat kampanye (gunakan solusinya)

Sebelum Anda bisa mendapatkan rekomendasi, Anda harus menerapkan versi solusi. Menyebarkan solusi juga dikenal sebagai membuat kampanye. Setelah Anda membuat kampanye, aplikasi klien Anda bisa mendapatkan rekomendasi menggunakan GetRecommendations API.

1. Buat kampanye dengan menjalankan perintah berikut. Berikan solusi versi ARN yang dikembalikan pada langkah sebelumnya. Untuk informasi selengkapnya tentang API, lihat CreateCampaign.

   ```
   aws personalize create-campaign \
     --name MovieRecommendationCampaign \
     --solution-version-arn arn:aws:personalize:us-west-2:acct-id:solution/
   MovieSolution/version-id \
     --min-provisioned-tps 1
   ```

   Respons sampel ditampilkan:

   ```
   {
     "campaignArn": "arn:aws:personalize:us-west-2:acct-id:campaign/
   MovieRecommendationCampaign"
   }
   ```

2. Periksa status penyebaran dengan menjalankan perintah berikut. Berikan kampanye ARN yang dikembalikan pada langkah sebelumnya. Untuk informasi selengkapnya tentang API, lihat DescribeCampaign.

```
aws personalize describe-campaign \
  --campaign-arn arn:aws:personalize:us-west-2:acct-id:campaign/
MovieRecommendationCampaign
```

Respons sampel ditampilkan:

```
{
  "campaign": {
      "name": "MovieRecommendationCampaign",
      "campaignArn": "arn:aws:personalize:us-west-2:acct-id:campaign/
MovieRecommendationCampaign",
      "solutionVersionArn": "arn:aws:personalize:us-west-2:acct-id:solution/
MovieSolution/<version-id>",
      "minProvisionedTPS": "1",
      "creationDateTime": 1543864775.923,
      "lastUpdatedDateTime": 1543864791.923,
      "status": "CREATE IN_PROGRESS"
  }
}
```

> **Note**
>
> Tunggu hingga `status` pertunjukan sebagai AKTIF sebelum mendapatkan rekomendasi dari kampanye.

Langkah 4: Dapatkan rekomendasi

Dapatkan rekomendasi dengan menjalankan `get-recommendations` perintah. Berikan kampanye ARN yang dikembalikan pada langkah sebelumnya. Dalam permintaan, Anda menentukan ID pengguna dari kumpulan data peringkat film. Untuk informasi selengkapnya tentang API, lihat GetRecommendations.

> **Note**
>
> Tidak semua resep mendukung `GetRecommendations` API. Untuk informasi selengkapnya, lihat Memilih resep.

AWS CLI Perintah yang Anda panggil dalam langkah inipersonalize-runtime,, berbeda dari pada langkah sebelumnya.

```
aws personalize-runtime get-recommendations \
  --campaign-arn arn:aws:personalize:us-west-2:acct-id:campaign/
MovieRecommendationCampaign \
  --user-id 123
```

Sebagai tanggapan, kampanye menampilkan daftar rekomendasi item (film IDs) yang mungkin disukai pengguna. Daftar ini diurutkan dalam urutan relevansi menurun bagi pengguna.

```
{
  "itemList": [
      {
          "itemId": "14"
      },
      {
          "itemId": "15"
      },
      {
          "itemId": "275"
      },
      {
          "itemId": "283"
      },
      {
          "itemId": "273"
      },
      ...
  ]
}
```

Memulai (SDK for Python (Boto3))

Tutorial ini menunjukkan cara menyelesaikan alur kerja Amazon Personalize dari awal hingga akhir dengan SDK for Python (Boto3).

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Topik

- Prasyarat
- Tutorial
- Memulai menggunakan Amazon Personalize APIs dengan notebook Jupyter (IPython)

Prasyarat

Berikut ini adalah langkah-langkah prasyarat untuk menggunakan contoh Python dalam panduan ini:

- Selesaikan Memulai prasyarat untuk mengatur izin yang diperlukan dan membuat data pelatihan. Jika Anda menggunakan data sumber Anda sendiri, pastikan bahwa data Anda diformat seperti prasyarat.
- Siapkan AWS SDK untuk Python (Boto3) lingkungan Anda seperti yang ditentukan dalamMenyiapkan AWS SDKs.

Tutorial

Dalam langkah-langkah berikut, Anda memverifikasi lingkungan Anda dan membuat SDK for Python (Boto3) klien untuk Amazon Personalize. Kemudian Anda mengimpor data, membuat dan menerapkan versi solusi dengan kampanye, dan mendapatkan rekomendasi.

Langkah 1: Verifikasi lingkungan Python Anda dan buat klien boto3

Setelah Anda menyelesaikan prasyarat, jalankan contoh Python berikut untuk mengonfirmasi bahwa lingkungan Anda dikonfigurasi dengan benar. Kode ini juga membuat klien Amazon Personalize boto3 yang Anda gunakan dalam tutorial ini. Jika lingkungan Anda dikonfigurasi dengan benar, daftar resep yang tersedia ditampilkan, dan Anda dapat menjalankan contoh lain dalam tutorial ini.

```
import boto3

personalizeRt = boto3.client('personalize-runtime')
personalize = boto3.client('personalize')

response = personalize.list_recipes()

for recipe in response['recipes']:
    print (recipe)
```

Langkah 2: Impor data

Setelah Anda membuat Amazon Personalisasi klien boto3 dan memverifikasi lingkungan Anda, impor data historis yang Anda buat saat Anda menyelesaikan. Memulai prasyarat Untuk mengimpor data historis ke Amazon Personalize, lakukan hal berikut:

1. Gunakan kode berikut untuk membuat skema di Amazon Personalize. Ganti `getting-started-schema` dengan nama untuk skema.

```
import json
schema = {
  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "ITEM_ID",
          "type": "string"
      },
      {
          "name": "TIMESTAMP",
          "type": "long"
      }
  ],
  "version": "1.0"
}

create_interactions_schema_response = personalize.create_schema(
    name='getting-started-schema',
    schema=json.dumps(schema)
)

interactions_schema_arn = create_interactions_schema_response['schemaArn']
print(json.dumps(create_interactions_schema_response, indent=2))
```

2. Buat grup dataset dengan kode berikut. Ganti `dataset group name` dengan nama untuk grup dataset.

```
response = personalize.create_dataset_group(name = 'dataset group name')
```

```
dataset_group_arn = response['datasetGroupArn']

description = personalize.describe_dataset_group(datasetGroupArn = dataset_group_arn)
['datasetGroup']

print('Name: ' + description['name'])
print('ARN: ' + description['datasetGroupArn'])
print('Status: ' + description['status'])
```

3. Buat kumpulan data interaksi Item di grup kumpulan data baru Anda dengan kode berikut. Beri nama kumpulan data dan berikan `schema_arn` dan `dataset_group_arn` dari langkah sebelumnya.

```
response = personalize.create_dataset(
    name = 'datase_name',
    schemaArn = 'schema_arn',
    datasetGroupArn = 'dataset_group_arn',
    datasetType = 'Interactions'
)

dataset_arn = response['datasetArn']
```

4. Impor data Anda dengan pekerjaan impor dataset dengan kode berikut. Kode menggunakan metode describe_dataset_import_job untuk melacak status pekerjaan.

   Teruskan parameter berikut sebagai parameter: nama untuk pekerjaan, `dataset_arn` dari langkah sebelumnya, jalur bucket Amazon S3 (`s3://bucket name/folder name/ratings.csv`) tempat Anda menyimpan data pelatihan, dan ARN peran layanan IAM Anda. Anda menciptakan peran ini sebagai bagian dariMemulai prasyarat. Amazon Personalize memerlukan izin untuk mengakses bucket. Lihat Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3.

```
import time
response = personalize.create_dataset_import_job(
    jobName = 'JobName',
    datasetArn = 'dataset_arn',
    dataSource = {'dataLocation':'s3://amzn-s3-demo-bucket/filename.csv'},
    roleArn = 'role_arn',
    importMode = 'FULL'
)

dataset_interactions_import_job_arn = response['datasetImportJobArn']
```

```
description = personalize.describe_dataset_import_job(
    datasetImportJobArn = dataset_interactions_import_job_arn)['datasetImportJob']

print('Name: ' + description['jobName'])
print('ARN: ' + description['datasetImportJobArn'])
print('Status: ' + description['status'])

max_time = time.time() + 3*60*60 # 3 hours
while time.time() < max_time:
    describe_dataset_import_job_response = personalize.describe_dataset_import_job(
        datasetImportJobArn = dataset_interactions_import_job_arn
    )
    status = describe_dataset_import_job_response["datasetImportJob"]['status']
    print("Interactions DatasetImportJob: {}".format(status))

    if status == "ACTIVE" or status == "CREATE FAILED":
        break

    time.sleep(60)
```

Langkah 3: Buat solusi

Setelah mengimpor data, Anda membuat solusi dan versi solusi sebagai berikut. Solusinya berisi konfigurasi untuk melatih model dan versi solusi adalah model terlatih.

1. Buat solusi baru dengan kode berikut. Lulus yang berikut sebagai parameter: `dataset_group_arn` dari sebelumnya, nama untuk solusi, dan ARN untuk resep User-Personalization-v 2 ()`arn:aws:personalize:::recipe/aws-user-personalization-v2`. Simpan ARN solusi baru Anda untuk digunakan nanti.

```
create_solution_response = personalize.create_solution(
  name='solution name',
  recipeArn= 'arn:aws:personalize:::recipe/aws-user-personalization-v2',
  datasetGroupArn = 'dataset group arn'
)
solution_arn = create_solution_response['solutionArn']
print('solution_arn: ', solution_arn)
```

2. Buat versi solusi dengan kode berikut. Lulus sebagai parameter `solution_arn` dari langkah sebelumnya. Kode berikut membuat versi solusi. Selama pelatihan, kode menggunakan

DescribeSolutionVersion operasi untuk mengambil status versi solusi. Saat pelatihan selesai, metode mengembalikan ARN versi solusi baru Anda. Simpan untuk digunakan nanti.

```
import time
import json

create_solution_version_response = personalize.create_solution_version(
    solutionArn = solution_arn
)

solution_version_arn = create_solution_version_response['solutionVersionArn']
print(json.dumps(create_solution_version_response, indent=2))

max_time = time.time() + 3*60*60 # 3 hours
while time.time() < max_time:
    describe_solution_version_response = personalize.describe_solution_version(
        solutionVersionArn = solution_version_arn
    )
    status = describe_solution_version_response["solutionVersion"]["status"]
    print("SolutionVersion: {}".format(status))

    if status == "ACTIVE" or status == "CREATE FAILED":
        break

    time.sleep(60)
```

Langkah 4: Buat kampanye

Setelah Anda membuat versi solusi, terapkan dengan kampanye Amazon Personalize. Gunakan kode berikut untuk membuat kampanye yang menerapkan versi solusi Anda. Lulus yang berikut ini sebagai parameter:`solution_version_arn`, dan nama untuk kampanye. Metode ini mengembalikan Nama Sumber Daya Amazon (ARN) kampanye baru Anda. Simpan untuk digunakan nanti.

```
response = personalize.create_campaign(
    name = 'campaign name',
    solutionVersionArn = 'solution version arn'
)

arn = response['campaignArn']
```

```
description = personalize.describe_campaign(campaignArn = arn)['campaign']
print('Name: ' + description['name'])
print('ARN: ' + description['campaignArn'])
print('Status: ' + description['status'])
```

Langkah 5: Dapatkan rekomendasi

Setelah Anda membuat kampanye, Anda dapat menggunakannya untuk mendapatkan rekomendasi. Kode berikut menunjukkan cara mendapatkan rekomendasi dari kampanye dan mencetak ID setiap item yang direkomendasikan. Lewati ARN dari kampanye yang Anda buat di langkah sebelumnya. Untuk ID pengguna, Anda meneruskan ID pengguna yang dari data pelatihan, seperti123.

```
response = personalizeRt.get_recommendations(
    campaignArn = 'Campaign ARN',
    userId = '123',
    numResults = 10
)

print("Recommended items")
for item in response['itemList']:
    print (item['itemId'])
```

Memulai menggunakan Amazon Personalize APIs dengan notebook Jupyter (IPython)

Untuk mulai menggunakan Amazon Personalize menggunakan notebook Jupyter, kloning atau unduh serangkaian notebook yang ditemukan di folder getting_started dari repositori Amazon Personalize samples. Notebook memandu Anda mengimpor data pelatihan, membuat solusi, membuat kampanye, dan mendapatkan rekomendasi menggunakan Amazon Personalize.

> **Note**
>
> Sebelum memulai dengan notebook, pastikan untuk membangun lingkungan Anda mengikuti langkah-langkah di README.md

Memulai (SDK for Java 2.x)

Tutorial ini menunjukkan kepada Anda cara menyelesaikan alur kerja Amazon Personalize dari awal hingga akhir dengan. AWS SDK for Java 2.x

Saat Anda menyelesaikan latihan memulai, untuk menghindari biaya yang tidak perlu, hapus sumber daya yang Anda buat. Untuk informasi selengkapnya, lihat Persyaratan untuk menghapus sumber daya Amazon Personalize.

Untuk contoh lainnya, lihat Selesaikan proyek Amazon Personalize.

Topik

- Prasyarat
- Selesaikan proyek Amazon Personalize

Prasyarat

Berikut ini adalah langkah-langkah prasyarat untuk menyelesaikan tutorial ini:

- SelesaikanMemulai prasyarat, untuk mengatur izin yang diperlukan dan membuat data pelatihan. Anda dapat menggunakan data sumber yang sama yang digunakan dalam Memulai (konsol) atau Memulai (AWS CLI) latihan. Jika Anda menggunakan data sumber Anda sendiri, pastikan bahwa data Anda diformat seperti dalam prasyarat.
- Siapkan lingkungan SDK for Java 2.x AWS dan kredensialnya seperti yang ditentukan dalam Menyiapkan prosedur di Panduan AWS SDK for Java 2.x Pengembang.AWS SDK for Java 2.x

Tutorial

Dalam langkah-langkah berikut, Anda menyiapkan project untuk menggunakan paket Amazon Personalize dan membuat Amazon Personalize SDK for Java 2.x client. Kemudian Anda mengimpor data, membuat dan menerapkan versi solusi dengan kampanye, dan mendapatkan rekomendasi.

Langkah 1: Siapkan project Anda untuk menggunakan paket Amazon Personalize

Setelah menyelesaikan prasyarat, tambahkan dependensi Amazon Personalize ke file pom.xml dan impor paket Amazon Personalize.

1. Tambahkan dependensi berikut ke file pom.xml Anda. Nomor versi terbaru mungkin berbeda dari kode contoh.

```
<dependency>
 <groupId>software.amazon.awssdk</groupId>
 <artifactId>personalize</artifactId>
 <version>2.16.83</version>
```

```
</dependency>
<dependency>
 <groupId>software.amazon.awssdk</groupId>
 <artifactId>personalizeruntime</artifactId>
 <version>2.16.83</version>
</dependency>
<dependency>
 <groupId>software.amazon.awssdk</groupId>
 <artifactId>personalizeevents</artifactId>
 <version>2.16.83</version>
</dependency>
```

2. Tambahkan pernyataan impor berikut ke proyek Anda.

```
// import client packages
import software.amazon.awssdk.services.personalize.PersonalizeClient;
import software.amazon.awssdk.services.personalizeruntime.PersonalizeRuntimeClient;
// Amazon Personalize exception package
import software.amazon.awssdk.services.personalize.model.PersonalizeException;
// schema packages
import software.amazon.awssdk.services.personalize.model.CreateSchemaRequest;
// dataset group packages
import software.amazon.awssdk.services.personalize.model.CreateDatasetGroupRequest;
import software.amazon.awssdk.services.personalize.model.DescribeDatasetGroupRequest;
// dataset packages
import software.amazon.awssdk.services.personalize.model.CreateDatasetRequest;
// dataset import job packages
import
 software.amazon.awssdk.services.personalize.model.CreateDatasetImportJobRequest;
import software.amazon.awssdk.services.personalize.model.DataSource;
import software.amazon.awssdk.services.personalize.model.DatasetImportJob;
import
 software.amazon.awssdk.services.personalize.model.DescribeDatasetImportJobRequest;
// solution packages
import software.amazon.awssdk.services.personalize.model.CreateSolutionRequest;
import software.amazon.awssdk.services.personalize.model.CreateSolutionResponse;
// solution version packages
import software.amazon.awssdk.services.personalize.model.DescribeSolutionRequest;
import
 software.amazon.awssdk.services.personalize.model.CreateSolutionVersionRequest;
import
 software.amazon.awssdk.services.personalize.model.CreateSolutionVersionResponse;
import
 software.amazon.awssdk.services.personalize.model.DescribeSolutionVersionRequest;
```

```
// campaign packages
import software.amazon.awssdk.services.personalize.model.CreateCampaignRequest;
import software.amazon.awssdk.services.personalize.model.CreateCampaignResponse;
// get recommendations packages
import
 software.amazon.awssdk.services.personalizeruntime.model.GetRecommendationsRequest;
import
 software.amazon.awssdk.services.personalizeruntime.model.GetRecommendationsResponse;
import software.amazon.awssdk.services.personalizeruntime.model.PredictedItem;
// Java time utility package
import java.time.Instant;
```

Langkah 2: Buat Amazon Personalisasi klien

Setelah menambahkan dependensi Amazon Personalize ke file pom.xml dan mengimpor paket yang diperlukan, buat klien Amazon Personalize berikut:

```
PersonalizeClient personalizeClient = PersonalizeClient.builder()
  .region(region)
  .build();

PersonalizeRuntimeClient personalizeRuntimeClient = PersonalizeRuntimeClient.builder()
  .region(region)
  .build();
```

Langkah 3: Impor data

Setelah Anda menginisialisasi klien Amazon Personalize, impor data historis yang Anda buat saat Anda menyelesaikan. Memulai prasyarat Untuk mengimpor data historis ke Amazon Personalize, lakukan hal berikut:

1. Simpan skema Avro berikut sebagai file JSON di direktori kerja Anda. Skema ini cocok dengan kolom dalam file CSV yang Anda buat saat Anda menyelesaikan. Memulai prasyarat

```
{
  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
```

```
            "type": "string"
        },
        {
            "name": "ITEM_ID",
            "type": "string"
        },
        {
            "name": "TIMESTAMP",
            "type": "long"
        }
    ],
    "version": "1.0"
}
```

2. Gunakan `createSchema` metode berikut untuk membuat skema di Amazon Personalize. Teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalize, nama untuk skema Anda, dan jalur file untuk file JSON skema yang Anda buat pada langkah sebelumnya. Metode ini mengembalikan Amazon Resource Name (ARN) skema baru Anda. Simpan untuk digunakan nanti.

```
    public static String createSchema(PersonalizeClient personalizeClient, String
schemaName, String filePath) {

        String schema = null;
        try {
            schema = new String(Files.readAllBytes(Paths.get(filePath)));
        } catch (IOException e) {
            System.out.println(e.getMessage());
        }

        try {
            CreateSchemaRequest createSchemaRequest = CreateSchemaRequest.builder()
                    .name(schemaName)
                    .schema(schema)
                    .build();

            String schemaArn =
personalizeClient.createSchema(createSchemaRequest).schemaArn();

            System.out.println("Schema arn: " + schemaArn);

            return schemaArn;
```

```
        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

3. Buat grup dataset. Gunakan `createDatasetGroup` metode berikut untuk membuat grup dataset. Teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalize dan nama untuk grup kumpulan data. Metode mengembalikan ARN dari grup dataset baru Anda. Simpan untuk digunakan nanti.

```
   public static String createDatasetGroup(PersonalizeClient personalizeClient,
String datasetGroupName) {

       try {
           CreateDatasetGroupRequest createDatasetGroupRequest =
CreateDatasetGroupRequest.builder()
                   .name(datasetGroupName)
                   .build();
           return
personalizeClient.createDatasetGroup(createDatasetGroupRequest).datasetGroupArn();
       } catch (PersonalizeException e) {
           System.out.println(e.awsErrorDetails().errorMessage());
       }
       return "";
   }
```

4. Buat kumpulan data interaksi Item. Gunakan `createDataset` metode berikut untuk membuat kumpulan data interaksi Item. Teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalize, nama untuk kumpulan data Anda, ARN skema Anda, ARN grup kumpulan data Anda, dan untuk jenis kumpulan data. `Interactions` Metode mengembalikan ARN dari dataset baru Anda. Simpan untuk digunakan nanti.

```
   public static String createDataset(PersonalizeClient personalizeClient,
           String datasetName,
           String datasetGroupArn,
           String datasetType,
           String schemaArn) {
       try {
           CreateDatasetRequest request = CreateDatasetRequest.builder()
                   .name(datasetName)
```

```
                    .datasetGroupArn(datasetGroupArn)
                    .datasetType(datasetType)
                    .schemaArn(schemaArn)
                    .build();

            String datasetArn = personalizeClient.createDataset(request)
                    .datasetArn();
            System.out.println("Dataset " + datasetName + " created.");
            return datasetArn;

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

5. Impor data Anda dengan pekerjaan impor dataset. Gunakan `createPersonalizeDatasetImportJob` metode berikut untuk membuat pekerjaan impor dataset.

   Teruskan parameter berikut ini sebagai parameter: klien layanan Amazon Personalisasi, nama untuk pekerjaan, ARN kumpulan data interaksi Item, jalur bucket Amazon S3 (`s3://`*`bucket name`*`/`*`folder name`*`/ratings.csv`) tempat Anda menyimpan data pelatihan, dan ARN peran layanan Anda (Anda membuat peran ini sebagai bagian dari). Memulai prasyarat Metode mengembalikan ARN dari pekerjaan impor dataset Anda. Secara opsional simpan untuk digunakan nanti.

```
    public static String createPersonalizeDatasetImportJob(PersonalizeClient
personalizeClient,
            String jobName,
            String datasetArn,
            String s3BucketPath,
            String roleArn) {

        long waitInMilliseconds = 60 * 1000;
        String status;
        String datasetImportJobArn;

        try {
            DataSource importDataSource = DataSource.builder()
                    .dataLocation(s3BucketPath)
```

```
                    .build();

            CreateDatasetImportJobRequest createDatasetImportJobRequest =
CreateDatasetImportJobRequest.builder()
                    .datasetArn(datasetArn)
                    .dataSource(importDataSource)
                    .jobName(jobName)
                    .roleArn(roleArn)
                    .build();

            datasetImportJobArn =
personalizeClient.createDatasetImportJob(createDatasetImportJobRequest)
                    .datasetImportJobArn();
            DescribeDatasetImportJobRequest describeDatasetImportJobRequest =
DescribeDatasetImportJobRequest.builder()
                    .datasetImportJobArn(datasetImportJobArn)
                    .build();

            long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            while (Instant.now().getEpochSecond() < maxTime) {

                DatasetImportJob datasetImportJob = personalizeClient
                        .describeDatasetImportJob(describeDatasetImportJobRequest)
                        .datasetImportJob();

                status = datasetImportJob.status();
                System.out.println("Dataset import job status: " + status);

                if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }
            return datasetImportJobArn;

        } catch (PersonalizeException e) {
            System.out.println(e.awsErrorDetails().errorMessage());
        }
        return "";
```

```
    }
```

Langkah 4: Buat solusi

Setelah mengimpor data, Anda membuat solusi dan versi solusi sebagai berikut. Solusinya berisi konfigurasi untuk melatih model dan versi solusi adalah model terlatih.

1. Buat solusi baru dengan `createPersonalizeSolution` metode berikut. Teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalize, kumpulan data Anda mengelompokkan Amazon Resource Name (ARN), nama untuk solusi, dan ARN untuk resep 2 (). User-Personalization-v `arn:aws:personalize:::recipe/aws-user-personalization-v2` Metode mengembalikan ARN solusi baru Anda. Simpan untuk digunakan nanti.

```
   public static String createPersonalizeSolution(PersonalizeClient
personalizeClient,
           String datasetGroupArn,
           String solutionName,
           String recipeArn) {

       try {
           CreateSolutionRequest solutionRequest = CreateSolutionRequest.builder()
                   .name(solutionName)
                   .datasetGroupArn(datasetGroupArn)
                   .recipeArn(recipeArn)
                   .build();

           CreateSolutionResponse solutionResponse =
personalizeClient.createSolution(solutionRequest);
           return solutionResponse.solutionArn();

       } catch (PersonalizeException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
           System.exit(1);
       }
       return "";
   }
```

2. Buat versi solusi dengan `createPersonalizeSolutionVersion` metode berikut. Lulus sebagai parameter ARN dari solusi langkah sebelumnya. Kode berikut pertama-tama memeriksa untuk melihat apakah solusi Anda sudah siap dan kemudian membuat versi solusi. Selama pelatihan, kode menggunakan DescribeSolutionVersion operasi untuk mengambil status versi

solusi. Saat pelatihan selesai, metode mengembalikan ARN versi solusi baru Anda. Simpan untuk digunakan nanti.

```
   public static String createPersonalizeSolutionVersion(PersonalizeClient
personalizeClient, String solutionArn) {
       long maxTime = 0;
       long waitInMilliseconds = 30 * 1000; // 30 seconds
       String solutionStatus = "";
       String solutionVersionStatus = "";
       String solutionVersionArn = "";

       try {
           DescribeSolutionRequest describeSolutionRequest =
DescribeSolutionRequest.builder()
                   .solutionArn(solutionArn)
                   .build();

           maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

           // Wait until solution is active.
           while (Instant.now().getEpochSecond() < maxTime) {

               solutionStatus =
personalizeClient.describeSolution(describeSolutionRequest).solution().status();
               System.out.println("Solution status: " + solutionStatus);

               if (solutionStatus.equals("ACTIVE") || solutionStatus.equals("CREATE
FAILED")) {
                   break;
               }
               try {
                   Thread.sleep(waitInMilliseconds);
               } catch (InterruptedException e) {
                   System.out.println(e.getMessage());
               }
           }

           if (solutionStatus.equals("ACTIVE")) {

               CreateSolutionVersionRequest createSolutionVersionRequest =
CreateSolutionVersionRequest.builder()
                       .solutionArn(solutionArn)
                       .build();
```

```
                CreateSolutionVersionResponse createSolutionVersionResponse =
personalizeClient
                        .createSolutionVersion(createSolutionVersionRequest);
                solutionVersionArn =
createSolutionVersionResponse.solutionVersionArn();

                System.out.println("Solution version ARN: " + solutionVersionArn);

                DescribeSolutionVersionRequest describeSolutionVersionRequest =
DescribeSolutionVersionRequest.builder()
                        .solutionVersionArn(solutionVersionArn)
                        .build();

                while (Instant.now().getEpochSecond() < maxTime) {

                    solutionVersionStatus =
personalizeClient.describeSolutionVersion(describeSolutionVersionRequest)
                            .solutionVersion().status();
                    System.out.println("Solution version status: " +
solutionVersionStatus);

                    if (solutionVersionStatus.equals("ACTIVE") ||
solutionVersionStatus.equals("CREATE FAILED")) {
                        break;
                    }
                    try {
                        Thread.sleep(waitInMilliseconds);
                    } catch (InterruptedException e) {
                        System.out.println(e.getMessage());
                    }
                }
                return solutionVersionArn;
            }
        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

Untuk informasi selengkapnya, lihat Membuat versi solusi secara manual. Saat Anda membuat versi solusi, Anda dapat mengevaluasi kinerjanya sebelum melanjutkan. Untuk informasi selengkapnya, lihat Mengevaluasi versi solusi Amazon Personalize dengan metrik.

Langkah 5: Buat kampanye

Setelah Anda melatih dan mengevaluasi versi solusi Anda, terapkan dengan kampanye Amazon Personalize. Gunakan `createPersonalCampaign` metode berikut untuk menerapkan versi solusi. Teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalisasi, Nama Sumber Daya Amazon (ARN) dari versi solusi yang Anda buat pada langkah sebelumnya, dan nama untuk kampanye. Metode mengembalikan ARN kampanye baru Anda. Simpan untuk digunakan nanti.

```
public static String createPersonalCompaign(PersonalizeClient personalizeClient, String
 solutionVersionArn, String name) {

    try {
        CreateCampaignRequest createCampaignRequest = CreateCampaignRequest.builder()
            .minProvisionedTPS(1)
            .solutionVersionArn(solutionVersionArn)
            .name(name)
            .build();

        CreateCampaignResponse campaignResponse =
 personalizeClient.createCampaign(createCampaignRequest);
        System.out.println("The campaign ARN is "+campaignResponse.campaignArn());
        return campaignResponse.campaignArn();

    } catch (PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
}
```

Untuk informasi selengkapnya tentang kampanye Amazon Personalisasi, lihat. Menerapkan versi solusi Amazon Personalize dengan kampanye

Langkah 6: Dapatkan rekomendasi

Setelah Anda membuat kampanye, Anda menggunakannya untuk mendapatkan rekomendasi. Gunakan `getRecs` metode berikut untuk mendapatkan rekomendasi bagi pengguna. Teruskan sebagai parameter klien runtime Amazon Personalisasi, Nama Sumber Daya Amazon (ARN)

kampanye yang Anda buat pada langkah sebelumnya, dan ID pengguna (misalnya,123) dari data historis yang Anda impor. Metode ini mencetak daftar item yang direkomendasikan ke layar.

```
   public static void getRecs(PersonalizeRuntimeClient personalizeRuntimeClient,
String campaignArn, String userId) {

       try {
           GetRecommendationsRequest recommendationsRequest =
GetRecommendationsRequest.builder()
                   .campaignArn(campaignArn)
                   .numResults(20)
                   .userId(userId)
                   .build();

           GetRecommendationsResponse recommendationsResponse =
personalizeRuntimeClient
                   .getRecommendations(recommendationsRequest);
           List<PredictedItem> items = recommendationsResponse.itemList();
           for (PredictedItem item : items) {
               System.out.println("Item Id is : " + item.itemId());
               System.out.println("Item score is : " + item.score());
           }

       } catch (AwsServiceException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
           System.exit(1);
       }
   }
```

Selesaikan proyek Amazon Personalize

Untuk all-in-one project yang menunjukkan cara menyelesaikan alur kerja Amazon Personalize dengan SDK for Java 2.x, lihat Amazon-Personalize-Java-App aktif. GitHub Proyek ini mencakup pelatihan beberapa versi solusi dengan resep yang berbeda, dan merekam acara dengan PutEvents operasi.

Untuk contoh tambahan, lihat kode yang ditemukan di folder personalisasi repositori contoh AWS SDK.

Mencocokkan kasus penggunaan Anda dengan Amazon Personalize resource

Rekomendasi Amazon Personalize dapat mengatasi kasus penggunaan berikut:

- Menghasilkan rekomendasi yang dipersonalisasi untuk pengguna
- Merekomendasikan item serupa atau terkait
- Merekomendasikan item yang sedang tren atau populer
- Merekomendasikan tindakan terbaik berikutnya untuk pengguna (hanya dengan sumber daya khusus)
- Mengurutkan ulang berdasarkan relevansi (hanya dengan sumber daya khusus)
- Menghasilkan segmen pengguna (hanya dengan sumber daya khusus)

Amazon Personalisasi menampilkan sumber daya berbasis domain dan sumber daya khusus yang dikonfigurasi untuk kasus penggunaan ini. Anda mulai dengan membuat grup dataset Domain atau grup kumpulan data Kustom:

- Dengan grup kumpulan data Domain, Anda membuat sumber daya yang telah dikonfigurasi sebelumnya dan dioptimalkan untuk domain VIDEO_ON_DEMAND atau ECOMMERCE.

  Jika Anda memiliki aplikasi streaming video atau e-commerce, kami sarankan Anda memulai dengan grup kumpulan data Domain. Anda masih dapat menambahkan sumber daya khusus, seperti solusi dan versi solusi yang dilatih untuk kasus penggunaan khusus. Dan Anda masih dapat menggunakan sumber daya khusus untuk mendapatkan rekomendasi batch. Anda tidak dapat membuat sumber daya tindakan terbaik berikutnya, termasuk kumpulan data Tindakan dan Interaksi Tindakan, dalam grup kumpulan data domain.
- Dengan grup kumpulan data Kustom, Anda memilih resep yang cocok dengan kasus penggunaan Anda. Anda kemudian melatih dan menerapkan hanya solusi dan versi solusi yang dapat dikonfigurasi (model rekomendasi Amazon Personalize terlatih). Saat siap, Anda dapat menerapkan versi solusi dalam kampanye untuk rekomendasi waktu nyata. Atau Anda bisa mendapatkan rekomendasi batch tanpa kampanye.

  Jika Anda tidak memiliki aplikasi streaming video atau e-commerce, kami sarankan Anda membuat grup dataset khusus. Jika tidak, mulailah dengan grup dataset Domain dan tambahkan sumber daya khusus seperlunya.

Bagian berikut memberikan informasi terperinci tentang kasus penggunaan dan resep khusus yang tersedia di Amazon Personalisasi. Jika Anda mencocokkan kasus penggunaan dengan sumber daya Amazon Personalize, perhatikan persyaratan datanya. Setelah Anda memilih kasus penggunaan atau resep, informasi ini dapat membantu saat Anda menyiapkan data AndaMempersiapkan data pelatihan untuk Amazon Personalize.

Topik

- Gunakan fitur kasing dan resep
- Memilih kasus penggunaan
- Memilih resep

Gunakan fitur kasing dan resep

Dengan beberapa kasus penggunaan dan resep, Amazon Personalize menggunakan fitur berikut untuk menghasilkan rekomendasi yang lebih relevan serta meningkatkan penemuan dan keterlibatan item.

Topik

- Personalisasi waktu nyata
- Eksplorasi
- Pembaruan otomatis

Personalisasi waktu nyata

Dengan beberapa kasus penggunaan dan resep, Amazon Personalize menggunakan personalisasi waktu nyata untuk memperbarui dan mengadaptasi rekomendasi sesuai dengan minat pengguna yang terus berkembang. Ini memperbarui rekomendasi untuk pengguna saat Anda merekam interaksi mereka dengan item atau tindakan yang ada pada pelatihan lengkap terbaru. Anda merekam interaksi ini dengan pelacak peristiwa dan PutEvents operasi atau, untuk interaksi dengan tindakan, PutActionInteractions operasi.

Untuk informasi selengkapnya tentang merekam acara, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi. Untuk informasi tentang data baru memengaruhi rekomendasi waktu nyata, termasuk personalisasi waktu nyata, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Kasus penggunaan dan resep berikut mendukung personalisasi waktu nyata:

- Direkomendasikan untuk Anda (kasus penggunaan ECOMMERCE)
- Pilihan teratas untuk Anda (kasus penggunaan VIDEO_ON_DEMAND)
- Resep personalisasi-v2 pengguna
- Resep Personalisasi Pengguna
- Resep peringkat-v2 yang dipersonalisasi
- Resep Peringkat yang Dipersonalisasi
- Next-Best-Action resep

Eksplorasi

Untuk beberapa kasus penggunaan domain dan resep khusus, Amazon Personalize menggunakan eksplorasi saat merekomendasikan item. Dengan eksplorasi, rekomendasi mencakup beberapa item atau tindakan yang biasanya cenderung tidak direkomendasikan bagi pengguna, seperti item atau tindakan baru, item atau tindakan dengan sedikit interaksi, atau item atau tindakan yang kurang relevan bagi pengguna berdasarkan perilaku mereka sebelumnya. Ini meningkatkan penemuan dan keterlibatan item ketika Anda memiliki katalog yang cepat berubah, atau ketika item baru, seperti artikel berita atau promosi, lebih relevan bagi pengguna karena masih segar.

Mengkonfigurasi eksplorasi

Jika Anda menggunakan resep User-Personalization-v 2, Amazon Personalize menangani konfigurasi eksplorasi untuk Anda dan item yang disertakan melalui eksplorasi memiliki `Exploration` respons rekomendasi`Reason`. Untuk memastikan item baru disertakan dalam rekomendasi, Anda dapat menggunakan filter promosi untuk mempromosikan item baru berdasarkan stempel waktu pembuatan. Untuk informasi lebih lanjut tentang promosi, lihatMempromosikan item dalam rekomendasi waktu nyata.

Untuk semua kasus penggunaan atau resep lain yang menggunakan eksplorasi, saat membuat pemberi rekomendasi atau kampanye khusus, atau saat membuat pekerjaan inferensi batch (sumber daya khusus), Anda dapat mengonfigurasi eksplorasi dengan bidang berikut:

- Penekanan pada penjelajahan item yang kurang relevan (bobot eksplorasi) - Konfigurasikan berapa banyak yang harus dijelajahi. Tentukan nilai desimal antara 0 hingga 1. Defaultnya adalah 0,3. Semakin dekat nilainya ke 1, semakin banyak eksplorasi. Dengan lebih banyak eksplorasi, rekomendasi mencakup lebih banyak item dengan lebih sedikit data interaksi item atau relevansi

berdasarkan perilaku sebelumnya. Pada nol, tidak ada eksplorasi terjadi dan rekomendasi didasarkan pada data saat ini (relevansi).

- Batas usia item eksplorasi - Tentukan usia item maksimum dalam beberapa hari sejak interaksi terbaru di semua item dalam kumpulan data interaksi item. Ini mendefinisikan ruang lingkup eksplorasi item berdasarkan usia item. Amazon Personalize menentukan usia item berdasarkan stempel waktu pembuatannya atau, jika data stempel waktu pembuatan tidak ada, data interaksi item. Untuk informasi selengkapnya cara Amazon Personalize menentukan usia item, lihat. Data stempel waktu pembuatan

  Untuk meningkatkan item yang dipertimbangkan Amazon Personalize selama eksplorasi, masukkan nilai yang lebih besar. Minimal adalah 1 hari dan defaultnya adalah 30 hari. Rekomendasi mungkin mencakup item yang lebih tua dari potongan usia item yang Anda tentukan. Ini karena item ini relevan dengan pengguna dan eksplorasi tidak mengidentifikasinya.

Gunakan kasing dan resep yang menggunakan eksplorasi

Untuk informasi selengkapnya tentang setiap kasus penggunaan atau resep yang menggunakan eksplorasi, lihat berikut ini:

- Direkomendasikan untuk Anda (kasus penggunaan ECOMMERCE)
- Pilihan teratas untuk Anda (kasus penggunaan VIDEO_ON_DEMAND)
- Resep personalisasi-v2 pengguna
- Resep Personalisasi Pengguna
- Next-Best-Action resep

Pembaruan otomatis

Untuk beberapa kasus penggunaan dan resep khusus, Amazon Personalize secara otomatis memperbarui versi pemberi rekomendasi atau solusi Anda untuk mempertimbangkan item atau tindakan baru untuk rekomendasi. Tidak ada biaya untuk pembaruan otomatis. Untuk daftar kasus penggunaan dan resep dengan pembaruan otomatis, lihatKasus penggunaan domain dan resep khusus dengan pembaruan otomatis.

Pembaruan otomatis berfungsi sebagai berikut:

- Saat Amazon Personalisasi secara otomatis memperbarui versi solusi atau pemberi rekomendasi Anda bergantung pada cara Anda mendapatkan rekomendasi:

  - Untuk rekomendasi real-time, Amazon Personalize memperbarui versi solusi atau pemberi rekomendasi setiap dua jam.
  - Untuk rekomendasi item batch, saat Anda membuat pekerjaan inferensi batch dan menentukan versi solusi terlatih lengkap terbaru untuk solusi Anda, Amazon Personalize secara otomatis memperbarui versi solusi untuk mempertimbangkan item baru selama eksplorasi. Jika Anda tidak menentukan versi solusi terbaru, tidak ada pembaruan yang terjadi.
- Dengan setiap pembaruan, Amazon Personalize mulai menyertakan item baru dalam rekomendasi yang digunakan. Eksplorasi Saat mempertimbangkan item atau tindakan baru, Amazon Personalize mempertimbangkan metadata apa pun untuk item tersebut. Namun, data ini akan memiliki efek yang lebih besar pada rekomendasi hanya setelah Anda merekam interaksi untuk item dan melatih ulang sepenuhnya.
- Agar pembaruan terjadi, Anda harus memberikan data tindakan, item, atau interaksi baru sejak pembaruan atau pelatihan ulang otomatis terakhir.
- Amazon Personalize mempertimbangkan item baru hingga Anda mengimpor 750.000 item. Ini adalah jumlah maksimum item yang dipertimbangkan selama pelatihan.

Pedoman dan persyaratan tambahan untuk sumber daya khusus

Jika Anda menggunakan sumber daya khusus, berikut ini adalah pedoman dan persyaratan untuk pembaruan otomatis:

- Versi solusi Anda harus diterapkan dalam kampanye. Kampanye Anda secara otomatis menggunakan versi solusi yang diperbarui.
- Pembaruan otomatis tidak sama dengan pelatihan otomatis. Pembaruan otomatis tidak membuat versi solusi yang sama sekali baru. Dan model tidak belajar dari data terbaru Anda. Untuk mempertahankan solusi Anda, frekuensi pelatihan otomatis Anda harus tetap setidaknya setiap minggu.
- Setelah solusi Anda secara otomatis membuat versi solusi baru atau Anda membuat yang baru secara manual, Amazon Personalize tidak akan secara otomatis memperbarui versi solusi yang lebih lama, bahkan jika Anda menerapkannya dalam kampanye.
- Jika setiap dua jam tidak cukup sering, dengan User-Personalization Anda dapat secara manual membuat versi solusi dengan `trainingMode` set `UPDATE` untuk menyertakan item baru tersebut dalam rekomendasi. Ingatlah bahwa Amazon Personalize secara otomatis hanya memperbarui versi solusi terlatih terbaru Anda. Versi solusi yang diperbarui secara manual tidak akan diperbarui secara otomatis di masa mendatang. Jika solusi Anda menggunakan pelatihan otomatis,

pembaruan otomatis akan dilanjutkan untuk versi solusi berikutnya. Jika tidak, buat solusi baru secara manual dengan mode pelatihan yang disetel ke FULL dan terapkan dalam kampanye.

Kasus penggunaan domain dan resep khusus dengan pembaruan otomatis

Untuk informasi selengkapnya tentang setiap kasus penggunaan atau resep yang menampilkan pembaruan otomatis, lihat berikut ini:

- Direkomendasikan untuk Anda (kasus penggunaan ECOMMERCE)
- Pilihan teratas untuk Anda (kasus penggunaan VIDEO_ON_DEMAND)
- Resep personalisasi-v2 pengguna
- Resep Personalisasi Pengguna
- Next-Best-Action resep

Memilih kasus penggunaan

Saat Anda membuat pemberi rekomendasi dalam grup kumpulan data Domain, Anda menentukan kasus penggunaan. Amazon Personalize melatih model yang mendukung pemberi rekomendasi dengan konfigurasi terbaik untuk kasus penggunaan. Setiap domain memiliki kasus penggunaan yang berbeda. Misalnya, jika Anda menentukan VIDEO_ON_DEMAND untuk grup kumpulan data Domain Anda, hanya kasus penggunaan VIDEO_ON_DEMAND yang tersedia. Setiap kasus penggunaan memiliki persyaratan berbeda untuk mendapatkan rekomendasi. Beberapa kasus penggunaan memerlukan jenis acara tertentu. Anda bebas untuk memasukkan jenis acara tambahan.

Untuk semua kasus penggunaan, data interaksi Anda harus memiliki yang berikut:

- Minimal 1000 catatan interaksi item dari pengguna yang berinteraksi dengan item di katalog Anda. Interaksi ini dapat berasal dari impor massal, atau acara streaming, atau keduanya.
- Minimal 25 pengguna unik IDs dengan setidaknya dua interaksi item untuk masing-masing.

Untuk rekomendasi kualitas, kami sarankan Anda memiliki minimal 50.000 interaksi item dari setidaknya 1.000 pengguna dengan dua atau lebih interaksi item masing-masing.

Topik

- Kasus penggunaan VIDEO_ON_DEMAND

- Kasus penggunaan ECOMMERCE

Kasus penggunaan VIDEO_ON_DEMAND

Bagian berikut mencantumkan persyaratan dan Nama Sumber Daya Amazon (ARN) untuk setiap kasus penggunaan VIDEO_ON_DEMAND. Untuk semua kasus penggunaan, data interaksi Anda harus memiliki yang berikut:

- Minimal 1000 catatan interaksi item dari pengguna yang berinteraksi dengan item di katalog Anda. Interaksi ini dapat berasal dari impor massal, atau acara streaming, atau keduanya.
- Minimal 25 pengguna unik IDs dengan setidaknya dua interaksi item untuk masing-masing.

Untuk rekomendasi kualitas, kami sarankan Anda memiliki minimal 50.000 interaksi item dari setidaknya 1.000 pengguna dengan dua atau lebih interaksi item masing-masing.

> **Note**
>
> Jika Anda menggunakan CreateRecommender API, berikan ARN yang tercantum di sini untuk resep ARN.

Topik

- Karena Anda menonton X
- Lebih seperti X
- Paling populer
- Tren sekarang
- Pilihan teratas untuk Anda

Karena Anda menonton X

Dapatkan rekomendasi untuk video yang juga ditonton pengguna lain berdasarkan video yang Anda tentukan. Dengan kasus penggunaan ini, Amazon Personalize secara otomatis memfilter video yang ditonton pengguna berdasarkan userID yang Anda tentukan dan peristiwa. `Watch` Jika Anda menerapkan filter Anda sendiri, filter Anda diterapkan setelah video yang ditonton pengguna disaring.

Saat memfilter, Amazon Personalize mempertimbangkan paling banyak 100 interaksi item per pengguna per jenis peristiwa. Ini berlaku untuk filter otomatis atau kustom apa pun. Anda dapat menggunakan konsol Service Quotas untuk meminta kenaikan batas ini. Untuk informasi selengkapnya, lihat bagian Meminta peningkatan kuota pada Panduan Pengguna Service Quotas. Jika Anda tidak mengimpor interaksi item untuk pengguna selama tiga bulan, filter Anda tidak lagi mempertimbangkan data historis pengguna. Untuk mempertimbangkan data ini, Anda harus mengimpor seluruh riwayat peristiwa pengguna lagi.

- Resep ARN: `arn:aws:personalize:::recipe/aws-vod-because-you-watched-x`
- GetRecommendations Persyaratan API:

  `userId`: Diperlukan

  `itemId`: Diperlukan
- Kumpulan data yang digunakan saat pelatihan: Hanya kumpulan data interaksi Item (wajib)
- Jenis acara yang diperlukan: Minimal, 1000 `Watch` acara.

Lebih seperti X

Dapatkan rekomendasi untuk video yang mirip dengan video yang Anda tentukan. Dengan kasus penggunaan ini, Amazon Personalize secara otomatis memfilter video yang ditonton pengguna berdasarkan userID yang Anda tentukan dan peristiwa. `Watch` Jika Anda menerapkan filter Anda sendiri, filter Anda diterapkan setelah video yang ditonton pengguna disaring.

Saat memfilter, Amazon Personalize mempertimbangkan paling banyak 100 interaksi item per pengguna per jenis peristiwa. Ini berlaku untuk filter otomatis atau kustom apa pun. Anda dapat menggunakan konsol Service Quotas untuk meminta kenaikan batas ini. Untuk informasi selengkapnya, lihat bagian Meminta peningkatan kuota pada Panduan Pengguna Service Quotas. Jika Anda tidak mengimpor interaksi item untuk pengguna selama tiga bulan, filter Anda tidak lagi mempertimbangkan data historis pengguna. Untuk mempertimbangkan data ini, Anda harus mengimpor seluruh riwayat peristiwa pengguna lagi.

- Resep ARN: `arn:aws:personalize:::recipe/aws-vod-more-like-x`
- GetRecommendations Persyaratan API:

  `userId`: Diperlukan

  `itemId`: Diperlukan

- Kumpulan data yang digunakan saat pelatihan:
  - Interaksi (wajib)
  - Item (wajib)
- Jumlah acara yang diperlukan: Minimal, 1000 acara jenis apa pun.
- Jenis acara yang direkomendasikan: `Watch` dan `Click` acara.

Paling populer

Dapatkan rekomendasi untuk video yang telah ditonton oleh sebagian besar pengguna.

- Resep ARN: `arn:aws:personalize:::recipe/aws-vod-most-popular`
- GetRecommendationspersyaratan:

  `userId`: Diperlukan

  `itemId`: Tidak digunakan
- Kumpulan data yang digunakan saat pelatihan: Hanya kumpulan data interaksi Item (wajib)
- Jenis acara yang diperlukan: Minimal, 1000 `Watch` acara.

Tren sekarang

Dapatkan rekomendasi untuk video yang sedang tren. Video yang sedang tren adalah item yang dengan cepat menjadi lebih populer di kalangan pengguna Anda. Setiap dua jam, Amazon Personalize secara otomatis mengevaluasi data interaksi Anda dan mengidentifikasi item yang sedang tren.

- Resep ARN: `arn:aws:personalize:::recipe/aws-vod-trending-now`
- GetRecommendations Persyaratan API:

  `userId`: Diperlukan hanya jika Anda memfilter berdasarkan CurrentUser atau berdasarkan item yang berinteraksi dengan pengguna

  `itemId`: Tidak digunakan
- Kumpulan data yang digunakan saat pelatihan: Hanya kumpulan data interaksi Item (wajib)
- Jumlah acara yang diperlukan: Minimal, 1000 acara jenis apa pun.

Pilihan teratas untuk Anda

Dapatkan rekomendasi konten yang dipersonalisasi untuk pengguna yang Anda tentukan. Dengan kasus penggunaan ini, Amazon Personalize secara otomatis memfilter video yang ditonton pengguna berdasarkan userID yang Anda tentukan dan peristiwa. `Watch` Jika Anda menerapkan filter Anda sendiri, filter Anda diterapkan setelah video yang ditonton pengguna disaring.

Saat memfilter, Amazon Personalize mempertimbangkan paling banyak 100 interaksi item per pengguna per jenis peristiwa. Ini berlaku untuk filter otomatis atau kustom apa pun. Anda dapat menggunakan konsol Service Quotas untuk meminta kenaikan batas ini. Untuk informasi selengkapnya, lihat bagian Meminta peningkatan kuota pada Panduan Pengguna Service Quotas. Jika Anda tidak mengimpor interaksi item untuk pengguna selama tiga bulan, filter Anda tidak lagi mempertimbangkan data historis pengguna. Untuk mempertimbangkan data ini, Anda harus mengimpor seluruh riwayat peristiwa pengguna lagi.

Saat merekomendasikan item, kasus penggunaan real-time-personalizationdan eksplorasi ini digunakan. Dan itu menggunakan pembaruan otomatis untuk mempertimbangkan item baru untuk rekomendasi.

- Resep ARN: `arn:aws:personalize:::recipe/aws-vod-top-picks`
- GetRecommendations persyaratan:

  `userId`: Diperlukan

  `itemId`: Tidak digunakan
- Kumpulan data yang digunakan saat pelatihan:
  - Interaksi (wajib)
  - Item (opsional)
  - Pengguna (opsional)
- Jumlah acara yang diperlukan: Minimal, 1000 acara.
- Jenis acara yang direkomendasikan: `Click` dan `Watch` acara.
- Parameter konfigurasi eksplorasi: Saat membuat pemberi rekomendasi, Anda dapat mengonfigurasi eksplorasi dengan yang berikut ini.
  - Penekanan pada penjelajahan item yang kurang relevan (bobot eksplorasi) - Konfigurasikan berapa banyak yang harus dijelajahi. Tentukan nilai desimal antara 0 hingga 1. Defaultnya adalah 0,3. Semakin dekat nilainya ke 1, semakin banyak eksplorasi. Dengan lebih banyak eksplorasi, rekomendasi mencakup lebih banyak item dengan lebih sedikit data interaksi item

atau relevansi berdasarkan perilaku sebelumnya. Pada nol, tidak ada eksplorasi terjadi dan rekomendasi didasarkan pada data saat ini (relevansi).

- Batas usia item eksplorasi - Tentukan usia item maksimum dalam beberapa hari sejak interaksi terbaru di semua item dalam kumpulan data interaksi item. Ini mendefinisikan ruang lingkup eksplorasi item berdasarkan usia item. Amazon Personalize menentukan usia item berdasarkan stempel waktu pembuatannya atau, jika data stempel waktu pembuatan tidak ada, data interaksi item. Untuk informasi selengkapnya cara Amazon Personalize menentukan usia item, lihat. Data stempel waktu pembuatan

  Untuk meningkatkan item yang dipertimbangkan Amazon Personalize selama eksplorasi, masukkan nilai yang lebih besar. Minimal adalah 1 hari dan defaultnya adalah 30 hari. Rekomendasi mungkin mencakup item yang lebih tua dari potongan usia item yang Anda tentukan. Ini karena item ini relevan dengan pengguna dan eksplorasi tidak mengidentifikasinya.

Kasus penggunaan ECOMMERCE

Bagian berikut mencantumkan persyaratan dan Nama Sumber Daya Amazon (ARN) untuk setiap kasus penggunaan ECOMMERCE. Untuk semua kasus penggunaan, data interaksi Anda harus memiliki yang berikut:

- Minimal 1000 catatan interaksi item dari pengguna yang berinteraksi dengan item di katalog Anda. Interaksi ini dapat berasal dari impor massal, atau acara streaming, atau keduanya.
- Minimal 25 pengguna unik IDs dengan setidaknya dua interaksi item untuk masing-masing.

Untuk rekomendasi kualitas, kami sarankan Anda memiliki minimal 50.000 interaksi item dari setidaknya 1.000 pengguna dengan dua atau lebih interaksi item masing-masing.

> **Note**
>
> Jika Anda menggunakan CreateRecommender API, berikan ARN yang tercantum di sini untuk resep ARN.

Topik

- Paling banyak dilihat
- Penjual terbaik

- Sering dibeli bersama
- Pelanggan yang melihat X juga melihat
- Direkomendasikan untuk Anda

Paling banyak dilihat

Dapatkan rekomendasi untuk item populer berdasarkan berapa kali pelanggan Anda melihat item.

- Resep ARN: `arn:aws:personalize:::recipe/aws-ecomm-popular-items-by-views`
- GetRecommendations persyaratan:

  `userId`: Diperlukan

  `itemId`: Tidak digunakan

  `inputList`: NA
- Kumpulan data yang digunakan saat pelatihan: Hanya kumpulan data interaksi Item (wajib)
- Jenis acara yang diperlukan: Minimal, 1000 `View` acara.

Penjual terbaik

Dapatkan rekomendasi untuk item populer berdasarkan berapa kali pelanggan Anda membeli item.

- Resep ARN: `arn:aws:personalize:::recipe/aws-ecomm-popular-items-by-purchases`
- GetRecommendations persyaratan:

  `userId`: Diperlukan

  `itemId`: Tidak digunakan

  `inputList`: NA
- Kumpulan data yang digunakan saat pelatihan: Hanya kumpulan data interaksi Item (wajib)
- Jenis acara yang diperlukan: Minimal, 1000 `Purchase` acara.

Sering dibeli bersama

Dapatkan rekomendasi untuk barang yang sering dibeli pelanggan bersama dengan item yang Anda tentukan.

- Resep ARN: `arn:aws:personalize:::recipe/aws-ecomm-frequently-bought-together`
- GetRecommendations persyaratan:

  `userId`: Diperlukan hanya jika Anda memfilter berdasarkan CurrentUser

  `itemId`: Diperlukan

  `inputList`: NA
- Kumpulan data yang digunakan saat pelatihan: Hanya kumpulan data interaksi Item (wajib)
- Jenis acara yang diperlukan: Minimal, 1000 `Purchase` acara.

Pelanggan yang melihat X juga melihat

Dapatkan rekomendasi untuk item yang juga dilihat pelanggan berdasarkan item yang Anda tentukan. Dengan kasus penggunaan ini, Amazon Personalize secara otomatis memfilter item yang dibeli pengguna berdasarkan userID yang Anda tentukan dan peristiwa. `Purchase` Jika Anda menerapkan filter Anda sendiri, filter Anda diterapkan setelah item yang sudah dibeli pengguna disaring.

Saat memfilter, Amazon Personalize mempertimbangkan paling banyak 100 interaksi item per pengguna per jenis peristiwa. Ini berlaku untuk filter otomatis atau kustom apa pun. Anda dapat menggunakan konsol Service Quotas untuk meminta kenaikan batas ini. Untuk informasi selengkapnya, lihat bagian Meminta peningkatan kuota pada Panduan Pengguna Service Quotas. Jika Anda tidak mengimpor interaksi item untuk pengguna selama tiga bulan, filter Anda tidak lagi mempertimbangkan data historis pengguna. Untuk mempertimbangkan data ini, Anda harus mengimpor seluruh riwayat peristiwa pengguna lagi.

- Resep ARN: `arn:aws:personalize:::recipe/aws-ecomm-customers-who-viewed-x-also-viewed`
- GetRecommendations persyaratan:

  `userId`: Diperlukan

`itemId`: Diperlukan

`inputList`: NA

- Kumpulan data yang digunakan saat pelatihan: Hanya kumpulan data interaksi Item (wajib)
- Jenis acara yang diperlukan: Minimal, 1000 `View` acara.
- Jenis acara yang direkomendasikan: `Purchase` acara.

Direkomendasikan untuk Anda

Dapatkan rekomendasi yang dipersonalisasi untuk item berdasarkan pengguna yang Anda tentukan. Dengan kasus penggunaan ini, Amazon Personalize secara otomatis menyaring item yang dibeli pengguna berdasarkan userID yang Anda tentukan dan peristiwa. `Purchase` Jika Anda menerapkan filter Anda sendiri, filter Anda diterapkan setelah item yang sudah dibeli pengguna disaring.

Saat memfilter, Amazon Personalize mempertimbangkan paling banyak 100 interaksi item per pengguna per jenis peristiwa. Ini berlaku untuk filter otomatis atau kustom apa pun. Anda dapat menggunakan konsol Service Quotas untuk meminta kenaikan batas ini. Untuk informasi selengkapnya, lihat bagian Meminta peningkatan kuota pada Panduan Pengguna Service Quotas. Jika Anda tidak mengimpor interaksi item untuk pengguna selama tiga bulan, filter Anda tidak lagi mempertimbangkan data historis pengguna. Untuk mempertimbangkan data ini, Anda harus mengimpor seluruh riwayat peristiwa pengguna lagi.

Saat merekomendasikan item, kasus penggunaan real-time-personalizationdan eksplorasi ini digunakan. Dan itu menggunakan pembaruan otomatis untuk mempertimbangkan item baru untuk rekomendasi.

- Resep ARN: `arn:aws:personalize:::recipe/aws-ecomm-recommended-for-you`
- GetRecommendations persyaratan:

  `userId`: Diperlukan

  `itemId`: Tidak digunakan

  `inputList`: NA
- Kumpulan data yang digunakan saat pelatihan:
  - Interaksi (wajib)
  - Item (opsional)

  - Pengguna (opsional)
- Jumlah acara yang diperlukan: Minimal, 1000 acara.
- Jenis acara yang direkomendasikan: `View` dan `Purchase` acara.
- Parameter konfigurasi eksplorasi: Saat membuat pemberi rekomendasi, Anda dapat mengonfigurasi eksplorasi dengan yang berikut ini.
  - Penekanan pada penjelajahan item yang kurang relevan (bobot eksplorasi) - Konfigurasikan berapa banyak yang harus dijelajahi. Tentukan nilai desimal antara 0 hingga 1. Defaultnya adalah 0,3. Semakin dekat nilainya ke 1, semakin banyak eksplorasi. Dengan lebih banyak eksplorasi, rekomendasi mencakup lebih banyak item dengan lebih sedikit data interaksi item atau relevansi berdasarkan perilaku sebelumnya. Pada nol, tidak ada eksplorasi terjadi dan rekomendasi didasarkan pada data saat ini (relevansi).
  - Batas usia item eksplorasi - Tentukan usia item maksimum dalam beberapa hari sejak interaksi terbaru di semua item dalam kumpulan data interaksi item. Ini mendefinisikan ruang lingkup eksplorasi item berdasarkan usia item. Amazon Personalize menentukan usia item berdasarkan stempel waktu pembuatannya atau, jika data stempel waktu pembuatan tidak ada, data interaksi item. Untuk informasi selengkapnya cara Amazon Personalize menentukan usia item, lihat. Data stempel waktu pembuatan

    Untuk meningkatkan item yang dipertimbangkan Amazon Personalize selama eksplorasi, masukkan nilai yang lebih besar. Minimal adalah 1 hari dan defaultnya adalah 30 hari. Rekomendasi mungkin mencakup item yang lebih tua dari potongan usia item yang Anda tentukan. Ini karena item ini relevan dengan pengguna dan eksplorasi tidak mengidentifikasi mereka.

Memilih resep

Saat Anda membuat solusi khusus, Anda menentukan resep dan mengonfigurasi parameter pelatihan. Resep adalah algoritma Amazon Personalisasi yang disiapkan untuk kasus penggunaan tertentu. Amazon Personalize menyediakan resep, berdasarkan kasus penggunaan umum, untuk model pelatihan. Saat Anda membuat versi solusi untuk solusi tersebut, Amazon Personalize melatih model yang mendukung versi solusi berdasarkan resep dan konfigurasi pelatihan.

Amazon Personalisasi resep gunakan yang berikut ini selama pelatihan:

- Atribut yang telah ditentukan sebelumnya dari data Anda
- Transformasi fitur yang telah ditentukan

- Algoritma yang telah ditentukan
- Pengaturan parameter awal untuk algoritme

Untuk mengoptimalkan model Anda, Anda dapat mengganti banyak parameter ini saat Anda membuat solusi. Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Topik

- Amazon Personalisasi jenis resep berdasarkan kasus penggunaan
- Amazon Personalisasi resep
- Melihat resep Amazon Personalize yang tersedia
- Resep personalisasi-v2 pengguna
- Resep Personalisasi Pengguna
- Resep Trend-Now
- Resep Hitung Popularitas
- Resep peringkat-v2 yang dipersonalisasi
- Resep Peringkat yang Dipersonalisasi
- Resep Barang Serupa
- Resep SIMS
- Next-Best-Action resep
- Resep item-Affinity
- Item-Attribute-Affinity resep
- Resep HRNN Legacy

Amazon Personalisasi jenis resep berdasarkan kasus penggunaan

Untuk memilih resep Anda, pertama-tama pilih kasus penggunaan Anda dari yang berikut ini dan catat jenis resep yang sesuai.

- Merekomendasikan item untuk pengguna (resep USER_PERSONALIZATION)

  Untuk memberikan rekomendasi yang dipersonalisasi bagi pengguna Anda, latih model Anda dengan resep USER_PERSONALIZATION. Rekomendasi yang dipersonalisasi membantu mendorong keterlibatan dan konversi yang lebih baik.

- Item peringkat untuk pengguna (resep PERSONALIZED_RANKING)

  Untuk mempersonalisasi urutan daftar kurasi atau hasil penelusuran untuk pengguna Anda, latih model Anda dengan resep PERSONALIZED_RANKING. Resep PERSONALIZED_RANKING membuat daftar yang dipersonalisasi dengan memberi peringkat ulang kumpulan item masukan berdasarkan tingkat minat yang diprediksi untuk pengguna tertentu. Daftar yang dipersonalisasi meningkatkan pengalaman pelanggan dan meningkatkan loyalitas dan keterlibatan pelanggan.

- Merekomendasikan item yang sedang tren atau populer (resep POPULAR_ITEMS)

  Untuk merekomendasikan item yang sedang tren atau populer gunakan resep POPULAR_ITEMS. Anda dapat menggunakan POPULAR_ITEMS jika pelanggan Anda sangat menghargai apa yang berinteraksi dengan pengguna lain. Penggunaan umum termasuk merekomendasikan konten media sosial viral, artikel berita terbaru, atau video olahraga terbaru.

- Merekomendasikan item serupa (resep RELATED_ITEMS)

  Untuk merekomendasikan item serupa, seperti item yang sering dibeli bersama atau film yang juga ditonton pengguna lain, Anda harus menggunakan resep RELATED_ITEMS. Merekomendasikan item serupa dapat membantu pelanggan Anda menemukan item dan dapat meningkatkan tingkat konversi pengguna.

- Merekomendasikan tindakan terbaik berikutnya (resep PERSONALIZED_ACTIONS)

  Untuk merekomendasikan tindakan terbaik berikutnya bagi pengguna Anda secara real time, seperti mendaftar untuk program loyalitas Anda atau mengajukan permohonan kartu kredit, Anda harus menggunakan resep PERSONALIZED_ACTIONS. Merekomendasikan tindakan terbaik berikutnya dapat meningkatkan loyalitas pelanggan, menghasilkan lebih banyak pendapatan, dan meningkatkan pengalaman pengguna Anda.

- Mendapatkan segmen pengguna (resep USER_SEGMENTATION)

  Untuk mendapatkan segmen pengguna berdasarkan data input item, seperti pengguna yang kemungkinan besar akan berinteraksi dengan item dengan atribut tertentu, Anda harus menggunakan resep USER_SEGMENTATION. Mendapatkan segmen pengguna dapat membantu Anda membuat kampanye pemasaran lanjutan yang mempromosikan item yang berbeda ke segmen pengguna yang berbeda berdasarkan kemungkinan mereka akan mengambil tindakan.

Amazon Personalisasi resep

Amazon Personalize menyediakan jenis resep berikut. Selain perbedaan perilaku, masing-masing jenis memiliki persyaratan yang berbeda untuk mendapatkan rekomendasi, seperti yang ditunjukkan pada tabel berikut.

Jenis resep	Resep	API	Persyaratan API
USER_PERSONALISASI	Personalisasi pengguna-v2 Personalisasi Pengguna Resep HRNN (warisan) Resep HRNN-metadata (warisan) Resep HRNN-ColdStart (warisan)	GetRecommendations	`userId`: Diperlukan `itemId`: Tidak digunakan `inputList` : NA
POPULER_ITEM	Trend-Sekarang Hitungan Popularitas	GetRecommendations	`userId`: Diperlukan hanya jika Anda menerapkan filter yang membutuhkannya `itemId`: Tidak digunakan `inputList` : NA

Jenis resep	Resep	API	Persyaratan API
PERSONALIZED_RANKING	Personalisasi-peringkat-v2 Peringkat yang Dipersonalisasi	GetPersonalizedRanking	`userId`: Diperlukan `itemId`: NA `inputList` : daftar ItemId
RELATED_ITEMS	Barang Serupa SIMS	GetRecommendations	`userId`: Diperlukan hanya jika Anda menerapkan filter yang membutuhkannya `itemId`: Diperlukan `inputList` : NA
PERSONALIZED_ACTIONS	Tindakan Terbaik Berikutnya-	GetActionRecommendations	`userId`: Diperlukan `actionId`: Tidak digunakan `itemId`: Tidak digunakan `inputList` : NA

Jenis resep	Resep	API	Persyaratan API
USER_SEGMENTATION	Barang-Afinitas Item-Atribusi-Afinitas	CreateBatchSegmentJob	Untuk persyaratan alur kerja batch, lihatMendapatkan segmen pengguna dengan pekerjaan segmen batch.

Melihat resep Amazon Personalize yang tersedia

Untuk melihat daftar resep yang tersedia:

- Di konsol Amazon Personalize, pilih grup kumpulan data. Dari panel navigasi, pilih Solusi dan resep, dan pilih tab Resep.
- Dengan AWS SDK untuk Python (Boto3), panggil ListRecipes API.
- Dengan AWS CLI, gunakan perintah berikut.

```
aws personalize list-recipes
```

Untuk mendapatkan informasi tentang resep menggunakan SDK for Python (Boto3), hubungi API. DescribeRecipe Untuk mendapatkan informasi tentang resep menggunakan AWS CLI, gunakan perintah berikut.

```
aws personalize describe-recipe --recipe-arn recipe_arn
```

Resep personalisasi-v2 pengguna

Resep User-Personalization-v 2 (aws-user-personalization-v2) merekomendasikan item yang akan berinteraksi dengan pengguna berdasarkan preferensi mereka. Misalnya, Anda dapat menggunakan

User-Personalization-v 2 untuk menghasilkan rekomendasi film yang dipersonalisasi untuk aplikasi streaming, atau rekomendasi produk yang dipersonalisasi untuk aplikasi ritel. Kasus penggunaan lainnya termasuk membuat rekomendasi real-time untuk situs berita atau rekomendasi batch untuk kampanye pemasaran yang dipersonalisasi.

User-Personalization-v2 dapat melatih hingga 5 juta item dari interaksi Item dan kumpulan data Item. Dan itu menghasilkan rekomendasi yang lebih relevan dengan latensi lebih rendah daripadaPersonalisasi Pengguna.

Karena User-Personalization-v 2 merekomendasikan item yang paling relevan kepada pengguna berdasarkan data Anda, lebih sering merekomendasikan item yang ada dengan data interaksi. Untuk memastikan rekomendasi menyertakan item baru, Anda dapat menggunakan promosi yang menyertakan beberapa item berdasarkan stempel waktu pembuatan. Untuk informasi lebih lanjut tentang promosi, lihatMempromosikan item dalam rekomendasi waktu nyata.

Resep ini menggunakan arsitektur berbasis transformer untuk melatih model yang mempelajari konteks dan melacak hubungan dan pola dalam data Anda. Transformer adalah jenis arsitektur jaringan saraf yang mengubah atau mengubah urutan input menjadi urutan output. Untuk Amazon Personalize, urutan input adalah riwayat interaksi item pengguna dalam data Anda. Urutan output adalah rekomendasi pribadi mereka. Untuk informasi lebih lanjut tentang transformer, lihat Apa Itu Transformer Dalam Kecerdasan Buatan? di Hub Konsep Komputasi AWS Awan.

User-personalization-v2 menggunakan model harga yang berbeda dari resep lainnya. Untuk informasi selengkapnya tentang harga, lihat Harga Amazon Personalisasi.

Topik

- Fitur resep
- Kumpulan data yang diperlukan dan opsional
- Properti dan hiperparameter

Fitur resep

User-personalization-v2 menggunakan fitur resep Amazon Personalize berikut saat membuat rekomendasi item:

- Personalisasi waktu nyata - Dengan personalisasi waktu nyata, Amazon Personalisasi pembaruan dan menyesuaikan rekomendasi item sesuai dengan minat pengguna yang terus berkembang. Untuk informasi selengkapnya, lihat Personalisasi waktu nyata.

- Eksplorasi — Dengan eksplorasi, rekomendasi mencakup item dengan data interaksi yang lebih sedikit atau relevansi dengan pengguna. Dengan User-Personalization-v 2, Amazon Personalize menangani konfigurasi eksplorasi untuk Anda. Untuk memastikan rekomendasi menyertakan item baru, Anda dapat menggunakan promosi untuk menyertakan item baru berdasarkan stempel waktu pembuatannya. Untuk informasi lebih lanjut tentang promosi, lihatMempromosikan item dalam rekomendasi waktu nyata.
- Pembaruan otomatis - Dengan pembaruan otomatis, Amazon Personalize secara otomatis memperbarui model terbaru (versi solusi) setiap dua jam untuk mempertimbangkan item baru untuk rekomendasi. Untuk informasi selengkapnya, lihat Pembaruan otomatis.
- Metadata dengan rekomendasi — Dengan resep User-Personalization-v 2, jika Anda memiliki kumpulan data Item dengan minimal satu kolom metadata, kampanye secara otomatis memiliki opsi untuk menyertakan metadata item dengan hasil rekomendasi. Anda tidak mengaktifkan metadata secara manual untuk kampanye Anda. Anda dapat menggunakan metadata untuk memperkaya rekomendasi di antarmuka pengguna, seperti menambahkan genre film ke komidi putar. Untuk informasi selengkapnya, lihat Metadata item dalam rekomendasi.

Kumpulan data yang diperlukan dan opsional

Untuk menggunakan User-Personalization-v 2, Anda harus membuat kumpulan data interaksi Item dan mengimpor minimal 1000 interaksi item. Amazon Personalize menghasilkan rekomendasi terutama berdasarkan data interaksi item. Untuk informasi lebih lanjut, lihatData interaksi item. User-Personalization-v2 dapat melatih hingga 5 juta item di seluruh interaksi Item dan kumpulan data Item.

Dengan User-Personalization-v 2, Amazon Personalize dapat menggunakan data interaksi Item yang mencakup hal-hal berikut:

- Jenis peristiwa dan data nilai peristiwa - Amazon Personalize menggunakan data tipe peristiwa, seperti jenis peristiwa klik atau tonton, untuk mengidentifikasi maksud dan minat pengguna melalui pola apa pun dalam perilaku mereka. Selain itu, Anda dapat menggunakan jenis peristiwa dan data nilai acara untuk memfilter catatan sebelum pelatihan. Untuk informasi selengkapnya, lihat Jenis peristiwa dan data nilai acara.

> **Note**
>
> Dengan User-Personalization-v 2, biaya pelatihan Anda didasarkan pada data interaksi Anda sebelum memfilter berdasarkan jenis atau nilai acara. Untuk informasi selengkapnya tentang harga, lihat Harga Amazon Personalisasi.

- Metadata kontekstual — Metadata kontekstual adalah data interaksi yang Anda kumpulkan di lingkungan pengguna pada saat kejadian, seperti lokasi atau jenis perangkat mereka. Untuk informasi selengkapnya, lihat Metadata kontekstual.

Kumpulan data berikut bersifat opsional dan dapat meningkatkan rekomendasi:

- Kumpulan data pengguna — Amazon Personalize dapat menggunakan data dalam kumpulan data Pengguna Anda untuk lebih memahami pengguna Anda dan minat mereka. Anda juga dapat menggunakan data dalam kumpulan data Pengguna untuk memfilter rekomendasi. Untuk informasi tentang data pengguna yang dapat Anda impor, lihatMetadata pengguna.
- Kumpulan data item — Amazon Personalize dapat menggunakan data dalam kumpulan data Item Anda untuk mengidentifikasi koneksi dan pola dalam perilakunya. Ini membantu Amazon Personalisasi memahami pengguna Anda dan minat mereka. Anda juga dapat menggunakan data dalam kumpulan data Item untuk memfilter rekomendasi. Untuk informasi tentang data item yang dapat Anda impor, lihatMetadata barang.

Properti dan hiperparameter

Resep User-Personalization-v 2 memiliki sifat sebagai berikut:

- Nama – `aws-user-personalization-v2`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-user-personalization-v2`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-user-personalization-v2`

Untuk informasi selengkapnya, lihat Memilih resep.

Tabel berikut menjelaskan hiperparameter untuk resep User-Personalization-v 2. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Dengan User-Personalization-v 2, jika Anda mengaktifkan pelatihan otomatis, Amazon Personalize secara otomatis melakukan HPO setiap 90 hari. Tanpa pelatihan otomatis, tidak ada HPO yang terjadi.

Tabel memberikan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]

- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)

Nama	Penjelasan
Hiperparameter algoritma	
`apply_recency_bias`	Menentukan apakah model harus memberi bobot lebih pada data interaksi item terbaru dalam kumpulan data interaksi Item Anda. Data interaksi terbaru mungkin mencakup perubahan mendadak dalam pola yang mendasari peristiwa interaksi. Untuk melatih model yang memberi bobot lebih pada peristiwa baru-baru ini, atur `apply_recency_bias` ke `true`. Untuk melatih model yang sama-sama menimbang semua interaksi masa lalu, atur `apply_recency_bias` ke `false`. Nilai default: `true` Rentang: `true` atau `false` Jenis nilai: Boolean HPO dapat disetel: Tidak

Resep Personalisasi Pengguna

> **Important**
>
> Kami merekomendasikan menggunakan resep User-personalisasi-v2. Ini dapat mempertimbangkan hingga 5 juta item dengan pelatihan yang lebih cepat, dan menghasilkan rekomendasi yang lebih relevan dengan latensi yang lebih rendah.

Resep User-Personalization (aws-user-personalization) dioptimalkan untuk semua skenario rekomendasi yang dipersonalisasi. Ini memprediksi item yang kemungkinan besar akan berinteraksi

dengan pengguna. Anda dapat menggunakan Personalisasi Pengguna untuk menghasilkan rekomendasi film yang dipersonalisasi untuk aplikasi streaming atau rekomendasi produk yang dipersonalisasi untuk aplikasi ritel.

Dengan Personalisasi Pengguna, Amazon Personalize menghasilkan rekomendasi terutama berdasarkan data interaksi item pengguna dalam kumpulan data interaksi Item. Itu juga dapat menggunakan item dan metadata pengguna apa pun di kumpulan data Item dan Pengguna Anda. Untuk informasi selengkapnya tentang data yang digunakannya, lihatKumpulan data yang diperlukan dan opsional.

Topik

- Fitur resep
- Kumpulan data yang diperlukan dan opsional
- Properti dan hiperparameter
- Pelatihan dengan resep Personalisasi Pengguna (konsol)
- Pelatihan dengan resep User-Personalization (Python SDK)
- Mendapatkan rekomendasi dan merekam tayangan (SDK for Python (Boto3))
- Contoh notebook Jupyter

Fitur resep

Personalisasi Pengguna menggunakan fitur resep Amazon Personalize berikut saat membuat rekomendasi item:

- Personalisasi waktu nyata - Dengan personalisasi waktu nyata, Amazon Personalisasi pembaruan dan menyesuaikan rekomendasi item sesuai dengan minat pengguna yang terus berkembang. Untuk informasi selengkapnya, lihat Personalisasi waktu nyata.
- Eksplorasi — Dengan eksplorasi, rekomendasi mencakup item atau item baru dengan data interaksi yang lebih sedikit. Ini meningkatkan penemuan dan keterlibatan item saat Anda memiliki katalog yang berubah dengan cepat, atau ketika item baru, seperti artikel berita atau promosi, lebih relevan bagi pengguna saat baru. Untuk informasi lebih lanjut tentang eksplorasi, lihatEksplorasi.
- Pembaruan otomatis - Dengan pembaruan otomatis, Amazon Personalize secara otomatis memperbarui model terbaru (versi solusi) setiap dua jam untuk mempertimbangkan item baru untuk rekomendasi. Untuk informasi selengkapnya, lihat Pembaruan otomatis.

Kumpulan data yang diperlukan dan opsional

Untuk menggunakan Personalisasi Pengguna, Anda harus membuat kumpulan data interaksi Item dan mengimpor minimal 1000 interaksi item. Amazon Personalize menghasilkan rekomendasi terutama berdasarkan data interaksi item.

Dengan Personalisasi Pengguna, Amazon Personalize dapat menggunakan data interaksi Item yang mencakup hal-hal berikut:

- Jenis peristiwa dan data nilai peristiwa - Amazon Personalize menggunakan data tipe peristiwa, seperti jenis peristiwa klik atau tonton, untuk mengidentifikasi maksud dan minat pengguna melalui pola apa pun dalam perilaku mereka. Selain itu, Anda dapat menggunakan jenis peristiwa dan data nilai acara untuk memfilter catatan sebelum pelatihan. Untuk informasi selengkapnya, lihat Jenis peristiwa dan data nilai acara.
- Metadata kontekstual — Metadata kontekstual adalah data interaksi yang Anda kumpulkan di lingkungan pengguna pada saat kejadian, seperti lokasi atau jenis perangkat mereka. Untuk informasi selengkapnya, lihat Metadata kontekstual.
- Data tayangan — Tayangan adalah daftar item yang terlihat oleh pengguna saat berinteraksi dengan (diklik, ditonton, dibeli, dan sebagainya) item tertentu. Untuk informasi selengkapnya, lihat Data tayangan.

Kumpulan data berikut bersifat opsional dan dapat meningkatkan rekomendasi:

- Kumpulan data pengguna — Amazon Personalize dapat menggunakan data dalam kumpulan data Pengguna Anda untuk lebih memahami pengguna Anda dan minat mereka. Anda juga dapat menggunakan data dalam kumpulan data Pengguna untuk memfilter rekomendasi. Untuk informasi tentang data pengguna yang dapat Anda impor, lihatMetadata pengguna.
- Kumpulan data item — Amazon Personalize dapat menggunakan data dalam kumpulan data Item Anda untuk mengidentifikasi koneksi dan pola dalam perilakunya. Ini membantu Amazon Personalisasi memahami pengguna Anda dan minat mereka. Anda juga dapat menggunakan data dalam kumpulan data Item untuk memfilter rekomendasi. Untuk informasi tentang data item yang dapat Anda impor, lihatMetadata barang.

Properti dan hiperparameter

Resep User-Personalization memiliki properti berikut:

- Nama – `aws-user-personalization`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-user-personalization`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-user-personalization`

Untuk informasi selengkapnya, lihat Memilih resep.

Tabel berikut menjelaskan hyperparameters untuk resep User-Personalization. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Hyperparameter featurisasi mengontrol cara memfilter data yang akan digunakan dalam pelatihan. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Tabel memberikan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Dapatkah parameter berpartisipasi dalam HPO?

Nama	Penjelasan
Hiperparameter algoritma	
`hidden_dimension`	Jumlah variabel tersembunyi yang digunakan dalam model. Variabel tersembunyi membuat ulang riwayat pembelian pengguna dan statistik item untuk menghasilkan skor peringkat. Tentukan lebih banyak dimensi tersembunyi saat kumpulan data interaksi Item Anda menyertakan pola yang lebih rumit. Menggunakan dimensi yang lebih tersembunyi membutuhkan dataset yang lebih besar dan lebih banyak waktu untuk diproses. Untuk menentukan nilai terbaik, gunakan HPO. Untuk menggunakan HPO, atur `performHPO` ke `true` saat Anda memanggil CreateSolution dan CreateSolutionVersion operasi.

Nama	Penjelasan
	Nilai default: 149 Rentang: [32, 256] Jenis nilai: Integer HPO dapat disetel: Ya
`bptt`	Menentukan apakah akan menggunakan teknik propagasi balik melalui waktu. Propagasi balik melalui waktu adalah teknik yang memperbarui bobot dalam algoritme berbasis jaringan saraf berulang. Gunakan kredit jangka panjang `bptt` untuk menghubungkan hadiah yang tertunda ke acara awal. Misalnya, hadiah yang tertunda dapat berupa pembelian yang dilakukan setelah beberapa klik. Acara awal bisa menjadi klik awal. Bahkan dalam jenis acara yang sama, seperti klik, ada baiknya untuk mempertimbangkan efek jangka panjang dan memaksimalkan total hadiah. Untuk mempertimbangkan efek jangka panjang, gunakan `bptt` nilai yang lebih besar. Menggunakan `bptt` nilai yang lebih besar membutuhkan kumpulan data yang lebih besar dan lebih banyak waktu untuk diproses. Nilai default: 32 Rentang: [2, 32] Jenis nilai: Integer HPO dapat disetel: Ya

Nama	Penjelasan
`recency_mask`	Menentukan apakah model harus mempertimbangkan tren popularitas terbaru dalam kumpulan data interaksi Item. Tren popularitas terbaru mungkin termasuk perubahan mendadak dalam pola yang mendasari peristiwa interaksi. Untuk melatih model yang memberi bobot lebih pada peristiwa baru-baru ini, atur `recency_mask` ke`true`. Untuk melatih model yang sama-sama menimbang semua interaksi masa lalu, atur `recency_mask` ke`false`. Untuk mendapatkan rekomendasi yang baik menggunakan bobot yang sama, Anda mungkin memerlukan kumpulan data pelatihan yang lebih besar. Nilai default: `True` Rentang: `True` atau `False` Jenis nilai: Boolean HPO dapat disetel: Ya
Hiperparameter featurisasi	

Nama	Penjelasan
`min_user_history_length_percentile`	Persentil minimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `min_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang riwayat pendek. Pengguna dengan riwayat singkat sering menunjukkan pola berdasarkan popularitas item, bukan kebutuhan atau keinginan pribadi pengguna. Menghapusnya dapat melatih model dengan lebih fokus pada pola yang mendasarinya dalam data Anda. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna, menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menetapkan nilai yang mempertahankan mayoritas pengguna, tetapi menghapus kasus tepi. Misalnya, menyetel `min_user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0.0 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`max_user_history_length_percentile`	Persentil maksimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `max_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang sejarah panjang karena data untuk pengguna ini cenderung mengandung noise. Misalnya, robot mungkin memiliki daftar panjang interaksi otomatis. Menghapus pengguna ini membatasi kebisingan dalam pelatihan. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan mayoritas pengguna tetapi menghapus kasus tepi. Misalnya, menyetel `min_user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0,99 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak
Hiperparameter konfigurasi kampanye eksplorasi item	

Nama	Penjelasan
`exploration_weight`	Menentukan seberapa sering rekomendasi menyertakan item dengan data interaksi item yang lebih sedikit atau relevansi. Semakin dekat nilainya ke 1,0, semakin banyak eksplorasi. Pada nol, tidak ada eksplorasi terjadi dan rekomendasi didasarkan pada data saat ini (relevansi). Untuk mengetahui informasi selengkapnya, lihat the section called “CampaignConfig”. Nilai default: 0,3 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`exploration_item_age_cut_of f`	Tentukan usia item maksimum dalam beberapa hari sejak interaksi terbaru di semua item di Kumpulan data interaksi item. Ini mendefinisikan ruang lingkup eksplorasi item berdasarkan usia item. Amazon Personalize menentukan usia item berdasarkan stempel waktu pembuatannya atau, jika data stempel waktu pembuatan tidak ada, data interaksi item. Untuk informasi selengkapnya cara Amazon Personalize menentukan usia item, lihat. Data stempel waktu pembuatan Untuk meningkatkan item yang dipertimbangkan Amazon Personalize selama eksplorasi, masukkan nilai yang lebih besar. Minimal adalah 1 hari dan defaultnya adalah 30 hari. Rekomendasi mungkin mencakup item yang lebih tua dari potongan usia item yang Anda tentukan. Ini karena item ini relevan dengan pengguna dan eksplorasi tidak mengident ifikasinya. Nilai default: 30,0 Rentang: Pelampung positif Jenis nilai: Float HPO dapat disetel: Tidak

Pelatihan dengan resep Personalisasi Pengguna (konsol)

Untuk menggunakan resep User-Personalization untuk menghasilkan rekomendasi di konsol, pertama-tama latih versi solusi baru menggunakan resep. Kemudian terapkan kampanye menggunakan versi solusi dan gunakan kampanye untuk mendapatkan rekomendasi.

Melatih versi solusi baru dengan resep User-Personalization (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Buat grup kumpulan data khusus dengan skema baru dan unggah kumpulan data Anda dengan data tayangan. Secara opsional sertakan CREATION_TIMESTAMP dan Metadata teks tidak terstruktur data dalam kumpulan data Item Anda sehingga Amazon Personalize dapat menghitung usia item dengan lebih akurat dan mengidentifikasi item dingin.

   Untuk informasi selengkapnya tentang mengimpor data, lihatMengimpor data pelatihan ke Amazon Personalize dataset.
3. Pada halaman grup Dataset, pilih grup kumpulan data baru yang berisi kumpulan data atau kumpulan data dengan data tayangan.
4. Di panel navigasi, pilih Solusi dan resep dan pilih Buat solusi.
5. Pada halaman Buat solusi, untuk nama Solusi, masukkan nama solusi baru Anda.
6. Untuk tipe Solusi, pilih Rekomendasi item untuk mendapatkan rekomendasi item bagi pengguna Anda.
7. Untuk Resep, pilih aws-user-personalization. Bagian konfigurasi Solusi muncul menyediakan beberapa opsi konfigurasi.
8. Dalam konfigurasi Peristiwa, jika kumpulan data interaksi Item Anda memiliki EVENT_TYPE atau kolom EVENT_TYPE dan EVENT_VALUE, gunakan kolom Tipe peristiwa dan ambang nilai Peristiwa secara opsional untuk memilih data interaksi item yang digunakan Amazon Personalize saat melatih model. Untuk informasi selengkapnya, lihat Memilih data interaksi item yang digunakan untuk pelatihan.

   Jika Anda memiliki beberapa jenis acara dan menggunakan resep User-Personalization-v 2 atau Personalized-Ranking-v 2 resep, Anda juga dapat menentukan bobot yang berbeda untuk jenis yang berbeda. Misalnya, Anda dapat mengonfigurasi solusi untuk memberikan bobot lebih untuk membeli acara daripada peristiwa klik. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi dengan konfigurasi peristiwa.
9. Konfigurasikan hyperparameters secara opsional untuk solusi Anda. Untuk daftar properti resep Personalisasi Pengguna dan hiperparameter, lihat. Properti dan hiperparameter
10. Pilih Buat dan latih solusi untuk memulai pelatihan. Halaman Dashboard ditampilkan.

    Anda dapat menavigasi ke halaman detail solusi untuk melacak kemajuan pelatihan di bagian Versi solusi. Saat pelatihan selesai, statusnya Aktif.

Membuat kampanye dan mendapatkan rekomendasi (konsol)

Ketika status versi solusi Anda Aktif, Anda siap untuk membuat kampanye dan mendapatkan rekomendasi sebagai berikut:

1. Di halaman detail solusi atau halaman Kampanye, pilih Buat kampanye baru.
2. Pada halaman Buat kampanye baru, untuk detail Kampanye, berikan informasi berikut:
   - Nama kampanye: Masukkan nama kampanye. Teks yang Anda masukkan di sini muncul di dasbor Kampanye dan halaman detail.
   - Solusi: Pilih solusi yang baru saja Anda buat.
   - ID versi solusi: Pilih ID versi solusi yang baru saja Anda buat.
   - Transaksi minimum yang disediakan per detik: Tetapkan transaksi minimum yang disediakan per detik yang didukung Amazon Personalize. Untuk informasi lebih lanjut, lihat CreateCampaign operasi.
3. Untuk konfigurasi Kampanye, berikan informasi berikut:
   - Bobot eksplorasi: Konfigurasikan berapa banyak yang harus dijelajahi, di mana rekomendasi menyertakan item dengan data interaksi item yang lebih sedikit atau relevansi lebih sering, semakin banyak eksplorasi yang Anda tentukan. Semakin dekat nilainya ke 1, semakin banyak eksplorasi. Pada nol, tidak ada eksplorasi terjadi dan rekomendasi didasarkan pada data saat ini (relevansi).
   - Pemotongan usia item eksplorasi: Masukkan usia item maksimum, dalam beberapa hari sejak interaksi terbaru, untuk menentukan ruang lingkup eksplorasi item. Untuk menambah jumlah item yang dipertimbangkan Amazon Personalize selama eksplorasi, masukkan nilai yang lebih besar.

     Misalnya, jika Anda memasukkan 10, hanya item dengan data interaksi item dari 10 hari sejak interaksi terbaru dalam kumpulan data yang dipertimbangkan selama eksplorasi.

     > **Note**
     >
     > Rekomendasi mungkin termasuk item tanpa data interaksi item dari luar kerangka waktu ini. Ini karena item ini relevan dengan minat pengguna, dan eksplorasi tidak diperlukan untuk mengidentifikasi mereka.
4. Pilih Buat kampanye.

5. Pada halaman detail kampanye, ketika status kampanye Aktif, Anda dapat menggunakan kampanye untuk mendapatkan rekomendasi dan merekam tayangan. Untuk informasi selengkapnya, lihat Langkah 5: Dapatkan rekomendasi di “Memulai.”

   Amazon Personalize secara otomatis memperbarui versi solusi terbaru Anda setiap dua jam untuk menyertakan data baru. Kampanye Anda secara otomatis menggunakan versi solusi yang diperbarui. Untuk informasi selengkapnya, lihat Pembaruan otomatis.

   Untuk memperbarui kampanye secara manual, pertama-tama Anda membuat dan melatih versi solusi baru menggunakan konsol atau CreateSolutionVersion operasi, dengan `trainingMode` disetel ke`update`. Anda kemudian memperbarui kampanye secara manual di halaman Kampanye konsol atau dengan menggunakan UpdateCampaign operasi.

   > **Note**
   >
   > Amazon Personalize tidak secara otomatis memperbarui versi solusi yang Anda buat sebelum 17 November 2020.

Pelatihan dengan resep User-Personalization (Python SDK)

Ketika Anda telah membuat grup kumpulan data dan mengunggah kumpulan data Anda dengan data tayangan, Anda dapat melatih solusi dengan resep Personalisasi Pengguna. Secara opsional sertakan CREATION_TIMESTAMP dan Metadata teks tidak terstruktur data dalam kumpulan data Item Anda sehingga Amazon Personalize dapat menghitung usia item dengan lebih akurat dan mengidentifikasi item dingin. Untuk informasi selengkapnya tentang membuat grup kumpulan data dan mengunggah data pelatihan, lihat. Membuat file JSON skema untuk skema Amazon Personalize

Untuk melatih solusi dengan resep User-Personalization menggunakan SDK AWS

1. Buat solusi baru menggunakan `create_solution` metode ini.

   Ganti `solution name` dengan nama solusi Anda dan `dataset group arn` dengan Amazon Resource Name (ARN) grup dataset Anda.

```
import boto3

personalize = boto3.client('personalize')

print('Creating solution')
```

```
create_solution_response = personalize.create_solution(name = 'solution name',
                           recipeArn = 'arn:aws:personalize:::recipe/aws-user-
personalization',
                           datasetGroupArn = 'dataset group arn',
                           )
solution_arn = create_solution_response['solutionArn']
print('solution_arn: ', solution_arn)
```

Untuk daftar properti aws-user-personalization resep dan hiperparameter, lihatProperti dan hiperparameter.

2. Buat versi solusi baru dengan data pelatihan yang diperbarui dan atur `trainingMode` untuk `FULL` menggunakan cuplikan kode berikut. Ganti `solution arn` dengan ARN solusi Anda.

```
import boto3

personalize = boto3.client('personalize')

create_solution_version_response = personalize.create_solution_version(solutionArn
 = 'solution arn',
                                                                       trainingMode='FULL')

new_solution_version_arn = create_solution_version_response['solutionVersionArn']
print('solution_version_arn:', new_solution_version_arn)
```

3. Setelah Amazon Personalize selesai membuat versi solusi, buat kampanye dengan parameter berikut:

   - Berikan yang baru `campaign name` dan yang `solution version arn` dihasilkan pada langkah 2.
   - Ubah hyperparameter konfigurasi eksplorasi `explorationWeight` item untuk mengonfigurasi berapa banyak yang harus dijelajahi. Item dengan data interaksi item yang lebih sedikit atau relevansi direkomendasikan lebih sering semakin dekat nilainya ke 1.0. Nilai defaultnya adalah 0,3.
   - Ubah parameter hyperparameter konfigurasi eksplorasi `explorationItemAgeCutOff` item untuk memberikan durasi maksimum, dalam beberapa hari relatif terhadap interaksi terbaru, yang itemnya harus dieksplorasi. Semakin besar nilainya, semakin banyak item yang dipertimbangkan selama eksplorasi.

Gunakan cuplikan Python berikut untuk membuat kampanye baru dengan penekanan pada eksplorasi dengan cut-off eksplorasi pada 30 hari. Membuat kampanye biasanya memakan waktu beberapa menit tetapi dapat memakan waktu lebih dari satu jam.

```
import boto3

personalize = boto3.client('personalize')

create_campaign_response = personalize.create_campaign(
    name = 'campaign name',
    solutionVersionArn = 'solution version arn',
    minProvisionedTPS = 1,
    campaignConfig = {"itemExplorationConfig": {"explorationWeight": "0.3",
 "explorationItemAgeCutOff": "30"}}
)

campaign_arn = create_campaign_response['campaignArn']
print('campaign_arn:', campaign_arn)
```

Dengan Personalisasi Pengguna, Amazon Personalize secara otomatis memperbarui versi solusi Anda setiap dua jam untuk menyertakan data baru. Kampanye Anda secara otomatis menggunakan versi solusi yang diperbarui. Untuk informasi selengkapnya, lihat Pembaruan otomatis.

Untuk memperbarui kampanye secara manual, pertama-tama Anda membuat dan melatih versi solusi baru menggunakan konsol atau CreateSolutionVersion operasi, dengan `trainingMode` disetel ke`update`. Anda kemudian memperbarui kampanye secara manual di halaman Kampanye konsol atau dengan menggunakan UpdateCampaign operasi.

> **Note**
>
> Amazon Personalize tidak secara otomatis memperbarui versi solusi yang Anda buat sebelum 17 November 2020.

Mendapatkan rekomendasi dan merekam tayangan (SDK for Python (Boto3))

Saat kampanye dibuat, Anda dapat menggunakannya untuk mendapatkan rekomendasi bagi pengguna dan merekam tayangan. Untuk informasi tentang mendapatkan rekomendasi batch menggunakan AWS SDKs lihatMembuat pekerjaan inferensi batch ()AWS SDKs.

Untuk mendapatkan rekomendasi dan merekam tayangan

1. Panggil metode `get_recommendations`. Ubah `campaign arn` ke ARN kampanye baru Anda dan `user id` ke userID pengguna.

```
import boto3

rec_response = personalize_runtime.get_recommendations(campaignArn = 'campaign
 arn', userId = 'user id')
print(rec_response['recommendationId'])
```

2. Buat pelacak acara baru untuk mengirim PutEvents permintaan. Ganti `event tracker name` dengan nama pelacak acara Anda dan `dataset group arn` dengan ARN grup dataset Anda.

```
import boto3

personalize = boto3.client('personalize')

event_tracker_response = personalize.create_event_tracker(
    name = 'event tracker name',
    datasetGroupArn = 'dataset group arn'
)
event_tracker_arn = event_tracker_response['eventTrackerArn']
event_tracking_id = event_tracker_response['trackingId']
print('eventTrackerArn:{},\n eventTrackingId:{}'.format(event_tracker_arn,
 event_tracking_id))
```

3. Gunakan `recommendationId` dari langkah 1 dan `event tracking id` dari langkah 2 untuk membuat `PutEvents` permintaan baru. Permintaan ini mencatat data tayangan baru dari sesi pengguna. Ubah `user id` ke ID pengguna.

```
import boto3

personalize_events.put_events(
```

```
    trackingId = 'event tracking id',
    userId= 'user id',
    sessionId = '1',
    eventList = [{
    'sentAt': datetime.now().timestamp(),
    'eventType' : 'click',
    'itemId' : rec_response['itemList'][0]['itemId'],
    'recommendationId': rec_response['recommendationId'],
    'impression': [item['itemId'] for item in rec_response['itemList']],
    }]
)
```

Contoh notebook Jupyter

Untuk contoh buku catatan Jupyter yang menunjukkan cara menggunakan resep Personalisasi Pengguna, lihat Personalisasi Pengguna dengan Eksplorasi.

Resep Trend-Now

Resep Trending-Now (aws-trending-now) menghasilkan rekomendasi untuk item yang dengan cepat menjadi lebih populer di kalangan pengguna Anda. Anda dapat menggunakan resep Trending-Now jika item yang semakin populer lebih relevan bagi pelanggan Anda. Misalnya, pelanggan Anda mungkin sangat menghargai apa yang berinteraksi dengan pengguna lain. Penggunaan umum termasuk merekomendasikan konten media sosial viral, artikel berita terbaru, atau video olahraga terbaru.

Trending-Now secara otomatis mengidentifikasi item tren teratas dengan menghitung peningkatan interaksi yang dimiliki setiap item selama interval waktu yang dapat dikonfigurasi. Item dengan tingkat kenaikan tertinggi dianggap sebagai item yang sedang tren. Waktu didasarkan pada data stempel waktu dalam kumpulan data interaksi Item Anda. Item yang dipertimbangkan berasal dari data interaksi yang Anda impor secara massal dan bertahap. Anda tidak perlu membuat versi solusi baru secara manual untuk Trending-Now untuk mempertimbangkan item baru dalam data interaksi.

Anda dapat menentukan interval waktu dengan memberikan `Trend discovery frequency` ketika Anda membuat solusi Anda. Misalnya, jika Anda menentukan `30 minutes` untuk`Trend discovery frequency`, untuk setiap 30 menit data, Amazon Personalize mengidentifikasi item dengan tingkat peningkatan interaksi terbesar sejak evaluasi terakhir. Frekuensi yang mungkin termasuk 30 menit, 1 jam, 3 jam, dan 1 hari. Pilih frekuensi yang selaras dengan distribusi data interaksi Anda. Data yang hilang selama interval yang Anda pilih dapat mengurangi akurasi

rekomendasi. Jika Anda mengimpor nol interaksi selama dua interval waktu terakhir, Amazon Personalize hanya merekomendasikan item populer, bukan item yang sedang tren.

Dengan Trending-Now, Anda memanggil GetRecommendations operasi atau mendapatkan rekomendasi di halaman kampanye Uji di konsol Amazon Personalize. Amazon Personalize mengembalikan item tren teratas. Anda lulus `userId` dalam permintaan Anda hanya jika Anda menerapkan filter yang membutuhkannya. Dengan GetRecommendations API, Anda dapat mengonfigurasi jumlah item yang sedang tren yang dikembalikan dengan `numResults` parameter. Anda tidak bisa mendapatkan rekomendasi batch dengan resep Trending-Now.

Untuk menggunakan Trending-Now, Anda harus membuat kumpulan data interaksi Item dengan setidaknya 1000 interaksi historis dan peristiwa unik yang digabungkan (setelah difilter oleh EventType dan, jika disediakan). eventValueThreshold Saat membuat rekomendasi item yang sedang tren, Trending-Now tidak menggunakan data dalam kumpulan data Item atau Pengguna. Namun, Anda masih dapat memfilter rekomendasi berdasarkan data dalam kumpulan data ini. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Topik

- Properti dan hiperparameter
- Membuat solusi (SDK untuk Python (Boto3))
- Contoh notebook Jupyter

Properti dan hiperparameter

Resep Trending-Now memiliki sifat-sifat berikut:

- Nama – `aws-trending-now`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-trending-now`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-trending-now-custom`

Untuk informasi selengkapnya, lihat Memilih resep.

Tabel berikut menjelaskan hyperparameters untuk resep Trending-Now. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Proses memilih nilai terbaik untuk

hyperparameter disebut optimasi hyperparameter (HPO). Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Dapatkah parameter berpartisipasi dalam HPO?

Nama	Penjelasan
Hiperparameter transformasi fitur	
`Trend discovery frequency`	Tentukan seberapa sering Amazon Personalize mengevaluasi data interaksi Anda dan mengidentifikasi item yang sedang tren. Misalnya, jika Anda menentuka n `30 minutesTrend discovery frequency` , setiap 30 menit Amazon Personalize mengidentifikasi item dengan tingkat peningkatan interaksi terbesar selama interval 30 menit. Frekuensi yang tersedia meliputi 30 menit, 1 jam, 2 jam, 3 jam, dan 1 hari. Pilih frekuensi yang selaras dengan distribusi data interaksi Anda. Data yang hilang selama interval yang Anda pilih dapat mengurangi akurasi rekomendasi. Jika Anda menggunakan operasi CreateSolution API dan tidak menentukan nilai, defaultnya adalah setiap 2 jam. **Note** Untuk nilai parameter apa pun yang lebih besar dari 2 jam, Amazon Personalize secara otomatis menyegarkan rekomendasi item yang sedang tren setiap 2 jam untuk memperhit ungkan interaksi baru dan item baru.

Nama	Penjelasan
	Nilai default: 2 jam Nilai yang mungkin: 30 menit, 1 jam, 2 jam, 3 jam, dan 1 hari. Jenis nilai: String HPO dapat disetel: Tidak

Membuat solusi (SDK untuk Python (Boto3))

Kode berikut menunjukkan cara membuat solusi dengan resep Trending-Now menggunakan SDK for Python (Boto3). Nilai yang mungkin untuk `trend_discovery_frequency` adalah`30 minutes`,`1 hour`,`3 hours`, dan`1 day`. Untuk informasi tentang membuat solusi dengan konsol, lihatMembuat solusi (konsol).

```
import boto3

personalize = boto3.client("personalize")

create_solution_response = personalize_client.create_solution(
    name="solution name",
    recipeArn="arn:aws:personalize:::recipe/aws-trending-now",
    datasetGroupArn="dataset group ARN",
    solutionConfig={
        "featureTransformationParameters": {
            "trend_discovery_frequency": "1 hour"
         }
    }
)
print(create_solution_response['solutionArn'])
```

Contoh notebook Jupyter

Untuk contoh notebook Jupyter yang menunjukkan cara menggunakan resep Trending-Now, lihat trending_now_example.ipynb di repositori Amazon Personalize samples. GitHub

Resep Hitung Popularitas

Popularity-Count merekomendasikan item paling populer berdasarkan data interaksi Anda. Item yang paling populer adalah item dengan data interaksi terbanyak dari pengguna unik. Resep mengembalikan item populer yang sama untuk semua pengguna. Popularity-Count adalah dasar yang baik untuk membandingkan dengan resep lain menggunakan metrik evaluasi yang dihasilkan Amazon Personalize saat Anda membuat versi solusi. Untuk informasi selengkapnya, lihat Mengevaluasi versi solusi Amazon Personalize dengan metrik.

Setelah Anda membuat versi solusi, pastikan Anda selalu memperbarui versi dan data solusi Anda. Dengan Popularity-Count, Anda harus secara manual membuat versi solusi baru (melatih ulang model) untuk Amazon Personalize untuk mempertimbangkan item baru untuk rekomendasi dan memperbarui model dengan perilaku terbaru pengguna Anda. Maka Anda harus memperbarui kampanye apa pun menggunakan versi solusi. Untuk informasi selengkapnya, lihat Mempertahankan relevansi rekomendasi.

Resep yang telah ditentukan ini memiliki sifat-sifat berikut:

- Nama – `aws-popularity-count`
- Resep ARN — `arn:aws:personalize:::recipe/aws-popularity-count`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-popularity-count`
- Transformasi fitur ARN — `arn:aws:personalize:::feature-transformation/sims`
- Jenis resep — `USER_PERSONALIZATION`

Popularity-Count tidak memiliki hiperparameter yang terbuka.

Resep peringkat-v2 yang dipersonalisasi

Resep Personalized-Ranking-v 2 menghasilkan peringkat item yang dipersonalisasi. Peringkat yang dipersonalisasi adalah daftar item yang direkomendasikan yang diberi peringkat ulang berdasarkan relevansi untuk pengguna tertentu. Ini berguna jika Anda memiliki koleksi item yang dipesan, seperti hasil penelusuran, promosi, atau daftar kurasi, dan Anda ingin memberikan peringkat ulang yang dipersonalisasi untuk setiap pengguna Anda.

Personalized-Ranking-v2 dapat melatih hingga 5 juta item dari interaksi Item dan kumpulan data Item. Dan itu menghasilkan peringkat yang lebih akurat dengan latensi yang lebih rendah daripada. Peringkat yang Dipersonalisasi

Saat Anda menggunakan Personalized-Ranking-v 2, Anda menentukan item yang akan diberi peringkat dalam operasi GetPersonalizedRanking API. Jika Anda menentukan item tanpa data interaksi, Amazon Personalize akan mengembalikan item ini tanpa skor rekomendasi dalam respons GetPersonalizedRanking API.

Resep ini menggunakan arsitektur berbasis transformer untuk melatih model yang mempelajari konteks dan melacak hubungan dan pola dalam data Anda. Transformer adalah jenis arsitektur jaringan saraf yang mengubah atau mengubah urutan input menjadi urutan output. Untuk Amazon Personalize, urutan input adalah riwayat interaksi item pengguna dalam data Anda. Urutan output adalah rekomendasi pribadi mereka. Untuk informasi lebih lanjut tentang transformer, lihat Apa Itu Transformer Dalam Kecerdasan Buatan? di Hub Konsep Komputasi AWS Awan.

Personalized-ranking-v2 menggunakan model harga yang berbeda dari resep lainnya. Untuk informasi selengkapnya tentang harga, lihat Harga Amazon Personalisasi.

Topik

- Fitur resep
- Kumpulan data yang diperlukan dan opsional
- Properti dan hiperparameter

Fitur resep

Personalized-ranking-v2 menggunakan fitur resep Amazon Personalize berikut saat memberi peringkat item:

- Personalisasi waktu nyata - Dengan personalisasi waktu nyata, Amazon Personalisasi pembaruan dan menyesuaikan rekomendasi item sesuai dengan minat pengguna yang terus berkembang. Untuk informasi selengkapnya, lihat Personalisasi waktu nyata.
- Metadata dengan rekomendasi — Dengan resep Personalized-Ranking-v 2, jika Anda memiliki kumpulan data Item dengan minimal satu kolom metadata, kampanye secara otomatis memiliki opsi untuk menyertakan metadata item dengan hasil rekomendasi. Anda tidak mengaktifkan metadata secara manual untuk kampanye Anda. Anda dapat menggunakan metadata untuk memperkaya rekomendasi di antarmuka pengguna, seperti menambahkan genre film ke komidi putar. Untuk informasi selengkapnya, lihat Metadata item dalam rekomendasi.

Kumpulan data yang diperlukan dan opsional

Untuk menggunakan Personalized-Ranking-v 2, Anda harus membuat kumpulan data interaksi Item dan mengimpor minimal 1000 interaksi item. Amazon Personalize menghasilkan peringkat terutama berdasarkan data interaksi item. Untuk informasi lebih lanjut, lihatData interaksi item. Personalized-Ranking-v2 dapat melatih hingga 5 juta item di seluruh interaksi Item dan kumpulan data Item.

Dengan Personalized-Ranking-v 2, Amazon Personalize dapat menggunakan data interaksi Item yang mencakup hal-hal berikut:

- Jenis peristiwa dan data nilai peristiwa — Amazon Personalize menggunakan data tipe peristiwa, seperti jenis peristiwa klik atau tonton, untuk mengidentifikasi maksud dan minat pengguna melalui pola apa pun dalam perilaku mereka. Selain itu, Anda dapat menggunakan jenis peristiwa dan data nilai acara untuk memfilter catatan sebelum pelatihan. Untuk informasi selengkapnya, lihat Jenis peristiwa dan data nilai acara.

  > **Note**
  >
  > Dengan Personalized-Ranking-v 2, biaya pelatihan Anda didasarkan pada data interaksi Anda sebelum memfilter berdasarkan jenis atau nilai acara. Untuk informasi selengkapnya tentang harga, lihat Harga Amazon Personalisasi.

- Metadata kontekstual — Metadata kontekstual adalah data interaksi yang Anda kumpulkan di lingkungan pengguna pada saat kejadian, seperti lokasi atau jenis perangkat mereka. Untuk informasi selengkapnya, lihat Metadata kontekstual.

Kumpulan data berikut bersifat opsional dan dapat meningkatkan rekomendasi:

- Kumpulan data pengguna — Amazon Personalize dapat menggunakan data dalam kumpulan data Pengguna Anda untuk lebih memahami pengguna Anda dan minat mereka. Anda juga dapat menggunakan data dalam kumpulan data Pengguna untuk memfilter rekomendasi. Untuk informasi tentang data pengguna yang dapat Anda impor, lihatMetadata pengguna.
- Kumpulan data item — Amazon Personalize dapat menggunakan data dalam kumpulan data Item Anda untuk mengidentifikasi koneksi dan pola dalam perilakunya. Ini membantu Amazon Personalisasi memahami pengguna Anda dan minat mereka. Anda juga dapat menggunakan data dalam kumpulan data Item untuk memfilter rekomendasi. Untuk informasi tentang data item yang dapat Anda impor, lihatMetadata barang.

Properti dan hiperparameter

Resep Personalized-Ranking-v 2 memiliki sifat sebagai berikut:

- Nama – `aws-personalized-ranking-v2`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-personalized-ranking-v2`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-personalized-ranking-v2`

Untuk informasi selengkapnya, lihat Memilih resep.

Tabel berikut menjelaskan hiperparameter untuk resep Personalized-Ranking-v 2. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Dengan Personalized-Ranking-v 2, jika Anda mengaktifkan pelatihan otomatis, Amazon Personalize secara otomatis melakukan HPO setiap 90 hari. Tanpa pelatihan otomatis, tidak ada HPO yang terjadi.

Tabel memberikan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)

Nama	Penjelasan
Hiperparameter algoritma	
`apply_recency_bias`	Menentukan apakah model harus memberi bobot lebih pada data interaksi item terbaru dalam kumpulan data interaksi Item Anda. Data interaksi terbaru mungkin mencakup perubahan mendadak dalam pola yang mendasari peristiwa interaksi. Untuk melatih model yang memberi bobot lebih pada peristiwa baru-baru ini, atur `apply_recency_bias` ke `true`. Untuk melatih model yang

Nama	Penjelasan
	sama-sama menimbang semua interaksi masa lalu, atur `apply_recency_bias` `kefalse`. Nilai default: `true` Rentang: `true` atau `false` Jenis nilai: Boolean HPO dapat disetel: Tidak

Resep Peringkat yang Dipersonalisasi

> **Important**
>
> Kami merekomendasikan menggunakan resep Personalized-ranking-v2. Ini dapat mempertimbangkan hingga 5 juta item dengan pelatihan yang lebih cepat, dan menghasilkan peringkat yang lebih akurat dengan latensi yang lebih rendah.

Resep Personalized-Ranking menghasilkan peringkat item yang dipersonalisasi. Peringkat yang dipersonalisasi adalah daftar item yang direkomendasikan yang diberi peringkat ulang untuk pengguna tertentu. Ini berguna jika Anda memiliki koleksi item yang dipesan, seperti hasil penelusuran, promosi, atau daftar kurasi, dan Anda ingin memberikan peringkat ulang yang dipersonalisasi untuk setiap pengguna Anda. Misalnya, dengan Personalized-Ranking, Amazon Personalize dapat memberi peringkat ulang hasil penelusuran yang Anda hasilkan. OpenSearch

Untuk melatih model, resep Peringkat yang Dipersonalisasi menggunakan data dalam kumpulan data interaksi Item Anda, dan jika Anda membuatnya, kumpulan data Item dan kumpulan data Pengguna di grup kumpulan data Anda (kumpulan data ini opsional). Dengan Personalized-Ranking, dataset Item Anda dapat disertakan Metadata teks tidak terstruktur dan kumpulan data interaksi Item dapat disertakan. Metadata kontekstual Untuk mendapatkan peringkat yang dipersonalisasi, gunakan GetPersonalizedRanking API.

Setelah Anda membuat versi solusi, pastikan Anda selalu memperbarui versi dan data solusi Anda. Dengan Personalized-Ranking, Anda harus secara manual membuat versi solusi baru (melatih ulang model) untuk Amazon Personalize untuk mempertimbangkan item baru untuk rekomendasi

dan memperbarui model dengan perilaku terbaru pengguna Anda. Maka Anda harus memperbarui kampanye apa pun menggunakan versi solusi. Untuk informasi selengkapnya, lihat Mempertahankan relevansi rekomendasi.

> **Note**
>
> Jika Anda memberikan item tanpa data interaksi untuk peringkat, Amazon Personalize akan mengembalikan item ini tanpa skor rekomendasi dalam respons GetPersonalizedRanking API.

Resep ini memiliki sifat sebagai berikut:

- Nama – `aws-personalized-ranking`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-personalized-ranking`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-personalized-ranking`
- Transformasi fitur ARN — `arn:aws:personalize:::feature-transformation/JSON-percentile-filtering`
- Jenis resep — `PERSONALIZED_RANKING`

Hyperparameter

Tabel berikut menjelaskan hyperparameters untuk resep Personalize-Ranking. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Hyperparameter featurisasi mengontrol cara memfilter data yang akan digunakan dalam pelatihan. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Dapatkah parameter berpartisipasi dalam optimasi hyperparameter (HPO)?

Nama	Penjelasan
Hiperparameter algoritma	
`hidden_dimension`	Jumlah variabel tersembunyi yang digunakan dalam model. Variabel tersembunyi membuat ulang riwayat pembelian pengguna dan statistik item untuk menghasilkan skor peringkat. Tentukan lebih banyak dimensi tersembunyi saat kumpulan data interaksi Item Anda menyertakan pola yang lebih rumit. Menggunakan dimensi yang lebih tersembunyi membutuhkan dataset yang lebih besar dan lebih banyak waktu untuk diproses. Untuk memutuskan nilai optimal, gunakan HPO. Untuk menggunakan HPO, atur `performHP0` ke `true` saat Anda memanggil CreateSolution dan CreateSolutionVersion operasi. Nilai default: 149 Rentang: [32, 256] Jenis nilai: Integer HPO dapat disetel: Ya
`bptt`	Menentukan apakah akan menggunakan teknik propagasi balik melalui waktu. Propagasi balik melalui waktu adalah teknik yang memperbarui bobot dalam algoritme berbasis jaringan saraf berulang. Gunakan kredit jangka panjang `bptt` untuk menghubungkan hadiah yang tertunda ke acara awal. Misalnya, hadiah yang tertunda dapat berupa pembelian yang dilakukan setelah beberapa klik. Peristiwa awal bisa menjadi klik awal. Bahkan dalam jenis acara yang sama, seperti klik, ada baiknya untuk mempertimbangkan efek jangka panjang dan memaksimalkan total hadiah. Untuk mempertimbangkan efek jangka panjang, gunakan `bptt` nilai yang lebih besar. Menggunakan

Nama	Penjelasan
	`bptt` nilai yang lebih besar membutuhkan kumpulan data yang lebih besar dan lebih banyak waktu untuk diproses. Nilai default: 32 Rentang: [2, 32] Jenis nilai: Integer HPO dapat disetel: Ya
`recency_mask`	Menentukan apakah model harus mempertimbangkan tren popularitas terbaru dalam kumpulan data interaksi Item. Tren popularitas terbaru mungkin termasuk perubahan mendadak dalam pola yang mendasari peristiwa interaksi. Untuk melatih model yang memberi bobot lebih pada peristiwa baru-baru ini, atur `recency_mask` ke `true`. Untuk melatih model yang sama-sama menimbang semua interaksi masa lalu, atur `recency_mask` ke `false`. Untuk mendapatkan rekomendasi yang baik menggunakan bobot yang sama, Anda mungkin memerlukan kumpulan data pelatihan yang lebih besar. Nilai default: `True` Rentang: `True` atau `False` Jenis nilai: Boolean HPO dapat disetel: Ya
Hiperparameter featurisasi	

Nama	Penjelasan
`min_user_history_length_percentile`	Persentil minimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `min_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang riwayat pendek. Pengguna dengan riwayat singkat sering menunjukkan pola berdasarkan popularitas item, bukan kebutuhan atau keinginan pribadi pengguna. Menghapusnya dapat melatih model dengan lebih fokus pada pola yang mendasarinya dalam data Anda. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna, menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menetapkan nilai yang mempertahankan mayoritas pengguna, tetapi menghapus kasus tepi. Misalnya, menyetel `min__user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0.0 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`max_user_history_length_percentile`	Persentil maksimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `max_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang sejarah panjang karena data untuk pengguna ini cenderung mengandung noise. Misalnya, robot mungkin memiliki daftar panjang interaksi otomatis. Menghapus pengguna ini membatasi kebisingan dalam pelatihan. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan mayoritas pengguna tetapi menghapus kasus tepi. Misalnya, menyetel `min__user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0,99 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Notebook sampel Peringkat yang Dipersonalisasi

Untuk contoh buku catatan Jupyter yang menunjukkan cara menggunakan resep Personalized-Ranking, lihat Personalisasi Contoh Peringkat.

Resep Barang Serupa

> **Note**
>
> Semua resep RELATED_ITEMS menggunakan data interaksi. Pilih Item Serupa jika Anda juga memiliki metadata item dan ingin Amazon Personalize menggunakannya untuk menemukan item serupa. Atau pilih Resep SIMS jika Anda ingin mengkonfigurasi lebih banyak hyperparameters untuk model.

Resep Similar-Items (aws-similar-items) menghasilkan rekomendasi untuk item yang mirip dengan item yang Anda tentukan. Gunakan Item Serupa untuk membantu pelanggan menemukan item baru di katalog Anda berdasarkan perilaku dan metadata item mereka sebelumnya. Merekomendasikan item serupa dapat meningkatkan keterlibatan pengguna, rasio klik-tayang, dan rasio konversi untuk aplikasi Anda.

Item Serupa menghitung kesamaan berdasarkan data interaksi dan metadata item apa pun yang Anda berikan. Ini memperhitungkan kemunculan bersama item dalam riwayat pengguna di kumpulan data Interaksi Anda, dan kesamaan metadata item apa pun. Misalnya, dengan Similar-Items, Amazon Personalize dapat merekomendasikan item yang sering dibeli pelanggan bersama dengan style Metadata kategoris () yang serupa, atau film yang juga ditonton pengguna lain dengan deskripsi serupa (). Metadata teks tidak terstruktur

Dengan Item Serupa, Anda memberikan ID item dalam GetRecommendations operasi (atau konsol Amazon Personalize) dan Amazon Personalize mengembalikan daftar item serupa. Atau Anda dapat menggunakan alur kerja batch untuk mendapatkan item serupa untuk semua item dalam inventaris Anda (lihatMendapatkan rekomendasi item batch). Saat Anda mendapatkan item serupa, Anda dapat memfilter item berdasarkan atribut item yang Anda tentukan dalam permintaan Anda. Anda melakukan ini dengan menambahkan`CurrentItem`. `attribute`elemen ke filter Anda. Sebagai contoh, lihat item data filter examples.

Untuk menggunakan Item Serupa, Anda harus membuat kumpulan data interaksi Item dengan setidaknya 1000 interaksi historis dan peristiwa unik (gabungan). Untuk prediksi yang lebih akurat, sebaiknya Anda juga membuat kumpulan data Item dan mengimpor metadata tentang item di katalog Anda. Item serupa tidak menggunakan data dalam kumpulan data Pengguna saat membuat rekomendasi. Anda masih dapat memfilter rekomendasi berdasarkan data dalam kumpulan data Pengguna. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Jika Anda memiliki kumpulan data Item dengan data tekstual dan data judul item, Anda dapat membuat tema untuk item terkait dalam rekomendasi batch. Untuk informasi selengkapnya, lihat Rekomendasi Batch dengan tema dari Content Generator

Anda bisa mendapatkan rekomendasi untuk item yang mirip dengan item dingin (item dengan interaksi kurang dari lima). Jika Amazon Personalize tidak dapat menemukan ID item yang Anda tentukan dalam permintaan rekomendasi atau file input batch, resep akan mengembalikan item populer sebagai rekomendasi.

Setelah Anda membuat versi solusi, pastikan Anda selalu memperbarui versi dan data solusi Anda. Dengan Similar-Items, Anda harus secara manual membuat versi solusi baru (melatih ulang model) untuk Amazon Personalize untuk mempertimbangkan item baru untuk rekomendasi dan memperbarui model dengan perilaku terbaru pengguna Anda. Maka Anda harus memperbarui kampanye apa pun menggunakan versi solusi. Untuk informasi selengkapnya, lihat Mempertahankan relevansi rekomendasi.

Properti dan hiperparameter

Resep Similar-Items memiliki sifat-sifat berikut:

- Nama – `aws-similar-items`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-similar-items`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-similar-items`

Untuk informasi selengkapnya, lihat Memilih resep.

Tabel berikut menjelaskan hiperparameter untuk resep Item Serupa. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Dapatkah parameter berpartisipasi dalam HPO?

Nama	Penjelasan
Hiperparameter algoritma	
`popularity_discount_factor`	Konfigurasikan bagaimana popularitas memengaruhi rekomendasi. Tentukan nilai yang mendekati nol untuk memasukkan item yang lebih populer. Tentukan nilai yang lebih dekat dengan nilai untuk kurang menekankan pada popularitas. Nilai default: 0.0 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak
`item_id_hidden_dim`	Jumlah variabel tersembunyi yang digunakan Amazon Personalize untuk memodelkan embeddings ID item berdasarkan data interaksi. Variabel tersembunyi membuat ulang riwayat pembelian pengguna dan statistik item untuk menghasilkan skor peringkat. Untuk menggunakannya`item_id_hidden_dim` , Anda harus menggunakan HPO dan memberikan nilai rentang minimum dan maksimum. Amazon Personalize menggunakan HPO untuk menemukan nilai terbaik dalam rentang yang Anda tentukan. Tentukan nilai maksimum yang lebih besar saat Anda memiliki kumpulan data interaksi Item yang besar. Menggunakan nilai maksimum yang lebih besar membutuhkan lebih banyak waktu untuk memproses. Untuk menggunakan HPO, atur `performHPO` ke `true` saat Anda memanggil CreateSolution operasi. Nilai default: 100 Rentang: [30, 200]

Nama	Penjelasan
	Jenis nilai: Integer HPO dapat disetel: Ya
`item_metadata_hidden_dim`	Jumlah variabel tersembunyi yang digunakan Amazon Personalize untuk memodelkan metadata item. Untuk menggunakannya `item_metadata_hidden_dim`, Anda harus menggunakan HPO dan memberikan nilai rentang minimum dan maksimum. Amazon Personalize menggunakan HPO untuk menemukan nilai terbaik dalam rentang yang Anda tentukan. Tentukan nilai maksimum yang lebih besar saat Anda memiliki kumpulan data interaksi Item yang besar. Menggunakan maksimum yang lebih besar membutuhkan lebih banyak waktu untuk memproses. Untuk menggunakan HPO, atur `performHPO` ke `true` saat Anda memanggil CreateSolution operasi. Nilai default: 100 Rentang: [30, 200] Jenis nilai: Integer HPO dapat disetel: Ya

Resep SIMS

> **Note**
>
> Semua resep RELATED_ITEMS menggunakan data interaksi. Pilih SIMS jika Anda ingin mengonfigurasi lebih banyak hyperparameters untuk model. Pilih metadata item Resep Barang Serupa jika Anda memiliki dan ingin Amazon Personalize menggunakannya untuk menemukan item serupa.

Resep Item-to-item persamaan (SIMS) menggunakan pemfilteran kolaboratif untuk merekomendasikan item yang paling mirip dengan item yang Anda tentukan saat Anda mendapatkan rekomendasi. SIMS menggunakan kumpulan data interaksi Item Anda, bukan metadata item seperti warna atau harga, untuk menentukan kesamaan. SIMS mengidentifikasi kemunculan bersama item dalam riwayat pengguna di kumpulan data Interaksi Anda untuk merekomendasikan item serupa. Misalnya, dengan SIMS Amazon Personalize dapat merekomendasikan item kedai kopi yang sering dibeli pelanggan bersama atau film yang juga ditonton pengguna yang berbeda.

Saat Anda mendapatkan rekomendasi item serupa, Anda dapat memfilter item berdasarkan atribut item yang Anda tentukan dalam permintaan Anda. Anda melakukan ini dengan menambahkan`CurrentItem`. `attribute`elemen ke filter Anda. Sebagai contoh, lihat item data filter examples.

Untuk menggunakan SIMS, Anda harus membuat kumpulan data interaksi Item dengan setidaknya 1000 interaksi historis dan peristiwa unik (gabungan). SIMS tidak menggunakan data dalam kumpulan data Pengguna atau Item saat membuat rekomendasi. Anda masih dapat memfilter rekomendasi berdasarkan data dalam kumpulan data ini. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Jika tidak ada data perilaku pengguna yang cukup untuk item atau ID item yang Anda berikan tidak ditemukan, SIMS merekomendasikan item populer. Setelah Anda membuat versi solusi, pastikan Anda selalu memperbarui versi dan data solusi Anda. Dengan SIMS, Anda harus secara manual membuat versi solusi baru (melatih ulang model) untuk Amazon Personalize untuk mempertimbangkan item baru untuk rekomendasi dan memperbarui model dengan perilaku terbaru pengguna Anda. Maka Anda harus memperbarui kampanye apa pun menggunakan versi solusi. Untuk informasi selengkapnya, lihat Mempertahankan relevansi rekomendasi.

Resep SIMS memiliki sifat-sifat berikut:

- Nama – `aws-sims`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-sims`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-sims`
- Transformasi fitur ARN — `arn:aws:personalize:::feature-transformation/sims`
- Jenis resep — `RELATED_ITEMS`

Tabel berikut menjelaskan hiperparameter untuk resep SIMS. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters

mengontrol bagaimana kinerja model. Hyperparameter featurisasi mengontrol cara memfilter data yang akan digunakan dalam pelatihan. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Dapatkah parameter berpartisipasi dalam optimasi hyperparameter (HPO)?

Nama	Penjelasan
Hiperparameter algoritma	
`popularity_discount_factor`	Konfigurasikan bagaimana popularitas memengaruhi rekomendasi. Tentukan nilai yang mendekati nol untuk memasukkan item yang lebih populer. Tentukan nilai yang lebih dekat dengan nilai untuk kurang menekankan pada popularitas. Nilai default: 0,5 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Ya
`min_cointeraction_count`	Jumlah minimum interaksi bersama yang Anda butuhkan untuk menghitung kesamaan antara sepasang item. Misalnya, nilai 3 berarti Anda memerlukan tiga atau lebih pengguna yang berinteraksi dengan kedua item agar algoritme menghitung kesamaannya. Nilai default: 3 Rentang: [0, 10]

Nama	Penjelasan
	Jenis nilai: Integer HPO dapat disetel: Ya
Hiperparameter featurisasi	
`min_user_history_length_percentile`	Persentil minimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data yang tersedia pada pengguna. Gunakan `min_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang riwayat pendek. Pengguna dengan riwayat singkat sering menunjukkan pola berdasarkan popularitas item, bukan kebutuhan atau keinginan pribadi pengguna. Menghapusnya dapat melatih model dengan lebih fokus pada pola yang mendasarinya dalam data Anda. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna, menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan mayoritas pengguna, tetapi menghapus kasus tepi. Nilai default: 0,005 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`max_user_history_length_percentile`	Persentil maksimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data yang tersedia pada pengguna. Gunakan `max_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang sejarah panjang. Pengguna dengan sejarah panjang cenderung mengandung kebisingan. Misalnya, robot mungkin memiliki daftar panjang interaksi otomatis. Menghapus pengguna ini membatasi kebisingan dalam pelatihan. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan mayoritas pengguna tetapi menghapus kasus tepi. Misalnya, `min_hist_length_percentile = 0.05` dan `max_hist_length_percentile = 0.95` mencakup semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0,995 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`min_item_interaction_count_percentile`	Persentil minimum interaksi item diperhitungkan untuk dimasukkan dalam pelatihan model. Gunakan `min_item_interaction_count_percentile` untuk mengecualikan persentase item dengan riwayat interaksi singkat. Item dengan sejarah singkat sering kali merupakan item baru. Menghapusnya dapat melatih model dengan lebih fokus pada item dengan riwayat yang diketahui. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna, menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan sebagian besar item, tetapi menghapus kasus tepi. Nilai default: 0,01 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`max_item_interaction_count_percentile`	Persentil maksimum interaksi item diperhitungkan untuk dimasukkan dalam pelatihan model. Gunakan `max_item_interaction_count_percentile` untuk mengecualikan persentase item dengan riwayat interaksi yang panjang. Item dengan sejarah panjang cenderung lebih tua dan mungkin sudah ketinggalan zaman. Misalnya, rilis film yang tidak dicetak. Menghapus item ini dapat fokus pada item yang lebih relevan. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menetapkan nilai yang mempertahankan sebagian besar item tetapi menghapus kasus tepi. Misalnya, `min_item_interaction_count_percentile = 0.05` dan `max_item_interaction_count_percentile = 0.95` mencakup semua item kecuali yang memiliki jumlah interaksi di bagian bawah atau atas 5%. Nilai default: 0,9 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Notebook sampel SIMS

Untuk contoh buku catatan Jupyter yang menunjukkan cara menggunakan resep SIMS, lihat Menemukan item serupa+ HPO.

Next-Best-Action resep

Resep Next-Best-Action (aws-next-best-action) menghasilkan rekomendasi waktu nyata untuk tindakan terbaik berikutnya bagi pengguna Anda. Tindakan terbaik berikutnya untuk pengguna adalah tindakan yang kemungkinan besar akan mereka ambil. Misalnya, mendaftar di program loyalitas Anda, mengunduh aplikasi Anda, atau mengajukan permohonan kartu kredit.

Dengan Next-Best-Action, Anda dapat memberikan rekomendasi tindakan yang dipersonalisasi untuk pengguna Anda saat mereka menggunakan aplikasi Anda. Menyarankan tindakan yang tepat untuk pengguna dapat mengakibatkan lebih banyak pengguna mengambil tindakan Anda. Bergantung pada tindakan yang ingin Anda rekomendasikan, Anda dapat meningkatkan loyalitas pelanggan, menghasilkan lebih banyak pendapatan, dan meningkatkan pengalaman pengguna aplikasi Anda. Untuk contoh kasus penggunaan yang menjelaskan bagaimana rekomendasi tindakan yang dipersonalisasi dapat bermanfaat bagi aplikasi e-commerce, lihatContoh kasus penggunaan.

Amazon Personalize memprediksi tindakan terbaik berikutnya dari tindakan yang Anda impor ke kumpulan data Tindakan. Ini mengidentifikasi tindakan yang kemungkinan besar akan diambil pengguna berdasarkan interaksi mereka dengan tindakan dan item. Jika data tindakan Anda menyertakan nilai tindakan, Amazon Personalize akan memperhitungkan nilai tindakan tersebut. Jika pengguna memiliki kemungkinan yang sama untuk mengambil dua tindakan berbeda, Amazon Personalize memberi peringkat tindakan dengan nilai yang lebih besar lebih tinggi.

Saat Anda mendapatkan rekomendasi tindakan real-time untuk pengguna, Amazon Personalize menampilkan daftar tindakan yang kemungkinan besar akan dilakukan pengguna dalam periode waktu yang dapat dikonfigurasi (the). `action optimization period` Misalnya, tindakan yang kemungkinan besar akan mereka ambil dalam 14 hari ke depan. Daftar ini diurutkan dalam urutan menurun berdasarkan skor kecenderungan. Skor ini mewakili kemungkinan bahwa pengguna akan mengambil tindakan.

Hingga Anda mengimpor data interaksi tindakan, Amazon Personalize merekomendasikan tindakan tanpa personalisasi, dan skor kecenderungan adalah 0,0. Suatu tindakan akan memiliki skor setelah tindakan memiliki yang berikut:

- Setidaknya 50 interaksi aksi dengan tipe acara TAKEN.
- Setidaknya 50 interaksi tindakan dengan jenis acara NOT_TAKEN atau VIEWED.

Interaksi aksi ini harus ada pada pelatihan versi solusi terbaru, dan harus terjadi dalam rentang waktu 6 minggu dari stempel waktu interaksi terbaru dalam kumpulan data interaksi Tindakan.

Untuk informasi selengkapnya tentang data yang digunakan Next-Best-Action resep, lihatKumpulan data yang diperlukan dan opsional.

Saat membuat solusi dengan Next-Best-Action resep, Anda dapat mengonfigurasi jendela waktu yang digunakan Amazon Personalize saat memprediksi tindakan dengan menggunakan hyperparameter featurisasi. `action optimization period` Untuk informasi selengkapnya, lihat Properti dan hiperparameter.

Topik

- Contoh kasus penggunaan
- Fitur resep
- Kumpulan data yang diperlukan dan opsional
- Properti dan hiperparameter

Contoh kasus penggunaan

Menyarankan tindakan yang tepat untuk pengguna dapat mengakibatkan lebih banyak pengguna mengambil tindakan Anda. Bergantung pada tindakan yang ingin Anda rekomendasikan, Anda berpotensi meningkatkan loyalitas pelanggan, menghasilkan lebih banyak pendapatan, dan meningkatkan pengalaman pengguna aplikasi Anda.

Misalnya, Anda mungkin memiliki aplikasi e-commerce yang menyarankan tindakan berbeda berikut:

- Berlangganan program loyalitas
- Unduh aplikasi seluler
- Pembelian dalam kategori Perhiasan
- Pembelian dalam kategori Kecantikan dan perawatan

Anda mungkin memiliki pengguna yang sering berbelanja di situs Anda dan telah berulang kali mengambil tindakan pembelian Perhiasan dan Kecantikan dan perawatan. Untuk pengguna ini, rekomendasi tindakan Amazon Personalisasi dan skornya mungkin termasuk yang berikut:

- Berlangganan program loyalitas

  Skor kecenderungan - 1.00
- Pembelian dalam kategori Perhiasan

Skor kecenderungan — 0.86

- Pembelian dalam kategori Kecantikan dan perawatan

  Skor kecenderungan — 0.85

Dengan rekomendasi tindakan ini, Anda tahu untuk meminta pengguna untuk mendaftar di program loyalitas Anda. Tindakan ini memiliki skor kecenderungan tertinggi dan itu adalah tindakan yang kemungkinan besar akan diambil pengguna. Ini karena pengguna sering berbelanja di toko Anda dan cenderung terlibat dengan manfaat dari program loyalitas Anda.

Fitur resep

Next-Best-ActionResepnya menggunakan fitur resep Amazon Personalize berikut saat membuat rekomendasi tindakan:

- Personalisasi waktu nyata: Amazon Personalize menggunakan personalisasi waktu nyata untuk memperbarui dan mengadaptasi rekomendasi tindakan sesuai dengan minat pengguna yang terus berkembang. Untuk informasi selengkapnya, lihat Personalisasi waktu nyata.
- Eksplorasi: Dengan eksplorasi, rekomendasi mencakup tindakan atau tindakan baru dengan data interaksi yang lebih sedikit. Untuk informasi lebih lanjut tentang eksplorasi, lihatEksplorasi.
- Pembaruan otomatis: Dengan pembaruan otomatis, Amazon Personalize secara otomatis memperbarui model terbaru (versi solusi) setiap dua jam untuk memasukkan tindakan baru dalam rekomendasi melalui eksplorasi. Untuk informasi selengkapnya, lihat Pembaruan otomatis.

Kumpulan data yang diperlukan dan opsional

Untuk menggunakan Next-Best-Action resep, Anda harus membuat kumpulan data berikut:

- Tindakan: Anda mengimpor data tentang tindakan Anda, seperti nilainya, ke dalam kumpulan data Amazon Personalize Actions.

  Dalam data tindakan, Anda dapat memberikan EXPIRATION_TIMESTAMP untuk setiap tindakan. Jika suatu tindakan telah kedaluwarsa, Amazon Personalize tidak akan memasukkannya ke dalam rekomendasi. Anda juga dapat memberikan REPEAT_FREQUENCY untuk setiap tindakan. Ini menunjukkan berapa lama Amazon Personalize harus menunggu sebelum merekomendasikan tindakan lagi setelah pengguna berinteraksi dengannya. Untuk informasi tentang data yang dapat disimpan oleh kumpulan data Tindakan, lihatMetadata aksi.

- Interaksi item: Kumpulan data interaksi Item Anda harus memiliki minimal 1000 interaksi item. Amazon Personalize menggunakan interaksi item untuk memahami status pengguna Anda saat ini dan minat mereka. Untuk informasi tentang data interaksi item, lihatData interaksi item.

Kumpulan data berikut adalah opsional:

- Kumpulan data interaksi tindakan: Interaksi tindakan adalah interaksi yang melibatkan pengguna dan tindakan dalam kumpulan data Tindakan Anda. Anda dapat mengimpor interaksi tindakan yang diambil, tidak diambil, dan dilihat. Meskipun data ini opsional, kami menyarankan Anda mengimpor data interaksi tindakan untuk rekomendasi kualitas. Jika Anda tidak memiliki data interaksi tindakan, Anda dapat membuat kumpulan data interaksi Tindakan kosong dan merekam interaksi pelanggan Anda dengan tindakan menggunakan operasi PutActionInteractions API.

  Hingga Anda mengimpor data interaksi tindakan, Amazon Personalize merekomendasikan tindakan tanpa personalisasi, dan skor kecenderungan adalah 0,0. Suatu tindakan akan memiliki skor setelah tindakan memiliki yang berikut:

  - Setidaknya 50 interaksi aksi dengan tipe acara TAKEN.
  - Setidaknya 50 interaksi tindakan dengan jenis acara NOT_TAKEN atau VIEWED.

  Interaksi aksi ini harus ada pada pelatihan versi solusi terbaru, dan harus terjadi dalam rentang waktu 6 minggu dari stempel waktu interaksi terbaru dalam kumpulan data interaksi Tindakan.

  Untuk informasi tentang data interaksi tindakan yang dapat Anda impor, lihatData interaksi aksi. Untuk informasi tentang merekam peristiwa interaksi tindakan, lihatMerekam peristiwa interaksi aksi waktu nyata.

  > **Note**
  >
  > Dengan Next-Best-Action, Amazon Personalize tidak menggunakan data tayangan atau metadata kontekstual dalam kumpulan data interaksi Tindakan.

- Pengguna: Amazon Personalize menggunakan data apa pun dalam kumpulan data Pengguna Anda untuk lebih memahami pengguna Anda dan minat mereka. Anda juga dapat menggunakan data dalam kumpulan data Pengguna untuk memfilter rekomendasi tindakan. Untuk informasi tentang data pengguna yang dapat Anda impor, lihatMetadata pengguna.
- Items: Amazon Personalize menggunakan data apa pun dalam kumpulan data Item beserta kumpulan data interaksi Item untuk mengidentifikasi koneksi dan pola dalam perilakunya. Ini

membantu Amazon Personalisasi memahami pengguna Anda dan minat mereka. Untuk informasi tentang data item yang dapat Anda impor, lihatMetadata barang.

Properti dan hiperparameter

Next-Best-ActionResepnya tidak mendukung optimasi hyperparameter. Next-Best-ActionResepnya memiliki sifat-sifat berikut:

- Nama – `aws-next-best-action`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-next-best-action`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-next-best-action`

Tabel berikut menjelaskan hiperparameter featurisasi untuk resep. aws-next-best-action Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Hyperparameter featurisasi mengontrol cara memfilter data yang akan digunakan dalam pelatihan.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Apakah parameter dapat berpartisipasi dalam HPO

Nama	Penjelasan
Hiperparameter featurisasi	
`action_optimization_period`	Jendela waktu yang digunakan Amazon Personalize saat memprediksi tindakan terbaik berikutnya untuk pengguna. Misalnya, tindakan yang kemungkinan besar akan dilakukan pengguna dalam 14 hari ke depan. Jika Anda tidak memiliki banyak data interaksi tindakan, tentukan nilai yang lebih besar. Jika Anda

Nama	Penjelasan
	tidak yakin nilai apa yang akan ditentukan, gunakan default. Nilai default: 14 Rentang: [7, 28] Jenis nilai: Integer HPO dapat disetel: Tidak

Resep item-Affinity

Resep Item-Affinity (aws-item-affinity) adalah resep USER_SEGMENTATION yang membuat segmen pengguna (grup pengguna) untuk setiap item yang Anda tentukan. Ini adalah pengguna Amazon Personalize memprediksi kemungkinan besar akan berinteraksi dengan setiap item. Gunakan Item-Affinity untuk mempelajari lebih lanjut tentang pengguna Anda dan mengambil tindakan berdasarkan segmen pengguna masing-masing.

Misalnya, Anda mungkin ingin membuat kampanye pemasaran untuk aplikasi ritel Anda berdasarkan preferensi pengguna untuk item di katalog Anda. Item-Affinity akan membuat segmen pengguna untuk setiap item berdasarkan data dalam kumpulan data Interaksi dan Item Anda. Anda dapat menggunakan ini untuk mempromosikan item yang berbeda ke segmen pengguna yang berbeda berdasarkan kemungkinan mereka akan mengambil tindakan (misalnya, mengklik item atau membeli item). Penggunaan lain mungkin termasuk produk penjualan silang ke berbagai pengguna atau mengidentifikasi calon pelamar kerja.

Untuk mendapatkan segmen pengguna berdasarkan item, Anda membuat solusi dan versi solusi dengan resep Item-Affinity, lalu tambahkan daftar item dalam format JSON ke bucket Amazon S3 dan buat pekerjaan segmen batch. Amazon Personalisasi menghasilkan segmen pengguna untuk setiap item ke lokasi keluaran Anda di Amazon S3. Data masukan Anda dapat memiliki maksimum 500 item untuk mendapatkan segmen pengguna. Untuk informasi tentang menyiapkan data masukan untuk pekerjaan segmen batch, lihatMempersiapkan data input untuk rekomendasi batch.

Anda harus memiliki kumpulan data interaksi Item untuk menggunakan Item-Affinity. Item dan kumpulan data Pengguna bersifat opsional. Anda bisa mendapatkan segmen pengguna dengan

pekerjaan segmen batch. Untuk informasi selengkapnya, lihat Mendapatkan segmen pengguna batch.

Setelah Anda membuat versi solusi, pastikan Anda selalu memperbarui versi dan data solusi Anda. Dengan Item-Affinity, Anda harus membuat versi solusi baru untuk Amazon Personalize untuk mempertimbangkan pengguna baru untuk segmen pengguna dan memperbarui model dengan perilaku terbaru pengguna Anda. Untuk mendapatkan segmen pengguna untuk suatu item, item tersebut harus ada saat Anda membuat versi solusi.

Resep Item-Affinity memiliki sifat-sifat berikut:

- Nama – `aws-item-affinity`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-item-affinity`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-item-affinity`
- Transformasi fitur ARN — `arn:aws:personalize:::feature-transformation/item-affinity`
- Jenis resep — `USER_SEGMENTATION`

Tabel berikut menjelaskan hiperparameter untuk resep Item-Affinity. Hyperparameter adalah parameter algoritma yang Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Anda tidak dapat menggunakan optimasi hyperparameter (HPO) dengan resep Item-Affinity.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)

Nama	Penjelasan
Hiperparameter algoritma	
`hidden_dimension`	Jumlah variabel tersembunyi yang digunakan dalam model. Variabel tersembunyi membuat ulang riwayat pembelian pengguna dan statistik item untuk menghasilkan skor peringkat. Tentukan lebih banyak

Nama	Penjelasan
	dimensi tersembunyi saat kumpulan data Interaksi Anda menyertakan pola yang lebih rumit. Menggunak an dimensi yang lebih tersembunyi membutuhkan dataset yang lebih besar dan lebih banyak waktu untuk diproses. Nilai default: 149 Rentang: [32, 256] Jenis nilai: Integer

Item-Attribute-Affinity resep

Resep Item-Attribute-Affinity (aws-item-attribute-affinity) adalah resep USER_SEGMENTATION yang membuat segmen pengguna (grup pengguna) untuk setiap atribut item yang Anda tentukan. Ini adalah pengguna Amazon Personalize memprediksi kemungkinan besar akan berinteraksi dengan item dengan atribut tertentu. Gunakan Item-Attribute-Affinity untuk mempelajari lebih lanjut tentang pengguna Anda dan mengambil tindakan berdasarkan segmen pengguna masing-masing.

Misalnya, Anda mungkin ingin membuat kampanye pemasaran untuk aplikasi ritel Anda berdasarkan preferensi pengguna untuk jenis sepatu di katalog Anda. Item-Attribute-Affinityakan membuat segmen pengguna untuk setiap data berbasis tipe sepatu di kumpulan data Interaksi dan Item Anda. Anda dapat menggunakan ini untuk mempromosikan sepatu yang berbeda ke segmen pengguna yang berbeda berdasarkan kemungkinan mereka akan mengambil tindakan (misalnya, mengklik sepatu atau membeli sepatu). Penggunaan lain mungkin termasuk mempromosikan genre film yang berbeda ke pengguna yang berbeda atau mengidentifikasi calon pelamar pekerjaan berdasarkan jenis pekerjaan.

Untuk mendapatkan segmen pengguna berdasarkan atribut item, Anda membuat solusi dan versi solusi dengan Item-Attribute-Affinity resep, lalu menambahkan daftar atribut item dalam format JSON ke bucket Amazon S3 dan membuat pekerjaan segmen batch. Amazon Personalisasi menghasilkan segmen pengguna untuk setiap item ke lokasi keluaran Anda di Amazon S3. Data masukan Anda dapat memiliki maksimal 10 kueri, di mana setiap kueri adalah satu atau lebih atribut item. Untuk informasi tentang menyiapkan data masukan untuk pekerjaan segmen batch, lihatMempersiapkan data input untuk rekomendasi batch.

Anda harus memiliki kumpulan data interaksi Item dan kumpulan data Item untuk digunakan. Item-Attribute-Affinity Kumpulan data Item Anda harus memiliki setidaknya satu kolom yang merupakan kolom metadata non-tekstual dan tidak dicadangkan. Anda bisa mendapatkan segmen pengguna dengan pekerjaan segmen batch. Untuk informasi selengkapnya, lihat Mendapatkan segmen pengguna batch dengan sumber daya khusus.

Setelah Anda membuat versi solusi, pastikan Anda selalu memperbarui versi dan data solusi Anda. Dengan Item-Attribute-Affinity, Anda harus membuat versi solusi baru untuk Amazon Personalize untuk mempertimbangkan pengguna baru untuk segmen pengguna dan memperbarui model dengan perilaku terbaru pengguna Anda. Untuk mendapatkan segmen pengguna untuk atribut item, atribut item harus ada saat Anda membuat versi solusi.

Item-Attribute-AffinityResepnya memiliki sifat-sifat berikut:

- Nama – `aws-item-attribute-affinity`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-item-attribute-affinity`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-item-attribute-affinity`
- Transformasi fitur ARN — `arn:aws:personalize:::feature-transformation/item-attribute-affinity`
- Jenis resep — `USER_SEGMENTATION`

Tabel berikut menjelaskan hiperparameter untuk Item-Attribute-Affinity resep. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Anda tidak dapat menggunakan optimasi hyperparameter (HPO) dengan resepnya. Item-Attribute-Affinity

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)

Nama	Penjelasan
Hiperparameter algoritma	

Nama	Penjelasan
`hidden_dimension`	Jumlah variabel tersembunyi yang digunakan dalam model. Variabel tersembunyi membuat ulang riwayat pembelian pengguna dan statistik item untuk menghasilkan skor peringkat. Tentukan lebih banyak dimensi tersembunyi saat kumpulan data Interaksi Anda menyertakan pola yang lebih rumit. Menggunakan dimensi yang lebih tersembunyi membutuhkan dataset yang lebih besar dan lebih banyak waktu untuk diproses. Nilai default: 149 Rentang: [32, 256] Jenis nilai: Integer

Resep HRNN Legacy

Resep HRNN Legacy tidak lagi tersedia. Dokumentasi ini untuk tujuan referensi.

Kami merekomendasikan penggunaan resep aws-user-personalizaton (User-Personalization) di atas resep HRNN warisan. Personalisasi Pengguna meningkatkan dan menyatukan fungsionalitas yang ditawarkan oleh resep HRNN. Untuk informasi selengkapnya, lihat Resep Personalisasi Pengguna.

Amazon Personalize dapat secara otomatis memilih resep hirarkis recurrent neural network (HRNN) yang paling tepat berdasarkan analisisnya terhadap data masukan. Opsi ini disebut AutoML. Untuk informasi selengkapnya, lihat Menggunakan AutoML untuk memilih resep HRNN (hanya API).

Topik

- Menggunakan AutoML untuk memilih resep HRNN (hanya API)
- Resep HRNN (warisan)
- Resep HRNN-metadata (warisan)
- Resep HRNN-ColdStart (warisan)

Menggunakan AutoML untuk memilih resep HRNN (hanya API)

Amazon Personalize dapat secara otomatis memilih resep hirarkis recurrent neural network (HRNN) yang paling tepat berdasarkan analisisnya terhadap data masukan. Opsi ini disebut AutoML. Untuk melakukan AutoML, atur `performAutoML` parameter `true` saat Anda memanggil API. CreateSolution

Anda juga dapat menentukan daftar resep yang diperiksa Amazon Personalize untuk menentukan resep optimal, berdasarkan metrik yang Anda tentukan. Dalam hal ini, Anda memanggil `CreateSolution` operasi, menentukan `true` `performAutoML` parameter, menghilangkan `recipeArn` parameter, dan menyertakan `solutionConfig` parameter, menentukan `metricName` dan `recipeList` sebagai bagian dari objek. `autoMLConfig`

Bagaimana resep dipilih ditunjukkan pada tabel berikut. Entah `performAutoML` atau `recipeArn` harus ditentukan tetapi tidak keduanya. AutoML hanya dilakukan dengan menggunakan resep HRNN.

performAutoML	recipeArn	solutionConfig	Hasil
true	menghilangkan	dihilangkan	Amazon Personalize memilih resepnya
true	menghilangkan	`autoMLConfig` : `metricName` dan `recipeList` ditentukan	Amazon Personalize memilih resep dari daftar yang mengoptimalkan metrik
menghilangkan	ditentukan	dihilangkan	Anda menentukan resepnya
menghilangkan	ditentukan	ditentukan	Anda menentukan resep dan mengganti properti pelatihan default

> **Note**
>
> `performAutoMLKapantrue`, semua parameter `solutionConfig` objek diabaikan kecuali untuk`autoMLConfig`.

Resep HRNN (warisan)

> **Note**
>
> Resep HRNN Legacy tidak lagi tersedia. Dokumentasi ini untuk tujuan referensi.
> Kami merekomendasikan penggunaan resep aws-user-personalizaton (User-Personalization) di atas resep HRNN warisan. Personalisasi Pengguna meningkatkan dan menyatukan fungsionalitas yang ditawarkan oleh resep HRNN. Untuk informasi selengkapnya, lihat Resep Personalisasi Pengguna.

Model resep Amazon Personalize hierarchical recurrent neural network (HRNN) mengubah perilaku pengguna untuk memberikan rekomendasi selama sesi. Sesi adalah serangkaian interaksi pengguna dalam jangka waktu tertentu dengan tujuan menemukan item tertentu untuk memenuhi kebutuhan, misalnya. Dengan menimbang interaksi terbaru pengguna lebih tinggi, Anda dapat memberikan rekomendasi yang lebih relevan selama sesi berlangsung.

HRNN mengakomodasi maksud dan minat pengguna, yang dapat berubah seiring waktu. Dibutuhkan riwayat pengguna yang dipesan dan secara otomatis menimbangnya untuk membuat kesimpulan yang lebih baik. HRNN menggunakan mekanisme gating untuk memodelkan bobot diskon sebagai fungsi yang dapat dipelajari dari item dan stempel waktu.

Amazon Personalize memperoleh fitur untuk setiap pengguna dari kumpulan data Anda. Jika Anda telah melakukan integrasi data real-time, fitur-fitur ini diperbarui secara real time sesuai dengan aktivitas pengguna. Untuk mendapatkan rekomendasi, Anda hanya memberikan`USER_ID`. Jika Anda juga menyediakan`ITEM_ID`, Amazon Personalize mengabaikannya.

Resep HRNN memiliki sifat-sifat berikut:

- Nama – `aws-hrnn`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-hrnn`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-hrnn`

- Transformasi fitur ARN — `arn:aws:personalize:::feature-transformation/JSON-percentile-filtering`
- Jenis resep — `USER_PERSONALIZATION`

Tabel berikut menjelaskan hiperparameter untuk resep HRNN. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Hyperparameter featurisasi mengontrol cara memfilter data yang akan digunakan dalam pelatihan. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Dapatkah parameter berpartisipasi dalam HPO?

Nama	Penjelasan
Hiperparameter algoritma	
`hidden_dimension`	Jumlah variabel tersembunyi yang digunakan dalam model. Variabel tersembunyi membuat ulang riwayat pembelian pengguna dan statistik item untuk menghasilkan skor peringkat. Tentukan lebih banyak dimensi tersembunyi saat kumpulan data interaksi Item Anda menyertakan pola yang lebih rumit. Menggunakan dimensi yang lebih tersembunyi membutuhkan dataset yang lebih besar dan lebih banyak waktu untuk diproses. Untuk memutuskan nilai optimal, gunakan HPO. Untuk menggunakan HPO, atur `performHPO` ke `true` saat Anda memanggil CreateSolution dan CreateSolutionVersion operasi. Nilai default: 43 Rentang: [32, 256]

Nama	Penjelasan
	Jenis nilai: Integer HPO dapat disetel: Ya
`bptt`	Menentukan apakah akan menggunakan teknik propagasi balik melalui waktu. Propagasi balik melalui waktu adalah teknik yang memperbarui bobot dalam algoritme berbasis jaringan saraf berulang. Gunakan kredit jangka panjang `bptt` untuk menghubungkan hadiah yang tertunda ke acara awal. Misalnya, hadiah yang tertunda dapat berupa pembelian yang dilakukan setelah beberapa klik. Peristiwa awal bisa menjadi klik awal. Bahkan dalam jenis acara yang sama, seperti klik, ada baiknya untuk mempertimbangkan efek jangka panjang dan memaksimalkan total hadiah. Untuk mempertimbangkan efek jangka panjang, gunakan `bptt` nilai yang lebih besar. Menggunakan `bptt` nilai yang lebih besar membutuhkan kumpulan data yang lebih besar dan lebih banyak waktu untuk diproses. Nilai default: 32 Rentang: [2, 32] Jenis nilai: Integer HPO dapat disetel: Ya

Nama	Penjelasan
`recency_mask`	Menentukan apakah model harus mempertimbangkan tren popularitas terbaru dalam kumpulan data interaksi Item. Tren popularitas terbaru mungkin termasuk perubahan mendadak dalam pola yang mendasari peristiwa interaksi. Untuk melatih model yang memberi bobot lebih pada peristiwa baru-baru ini, atur `recency_mask` ke`true`. Untuk melatih model yang sama-sama menimbang semua interaksi masa lalu, atur `recency_mask` ke`false`. Untuk mendapatkan rekomendasi yang baik menggunakan bobot yang sama, Anda mungkin memerlukan kumpulan data pelatihan yang lebih besar. Nilai default: `True` Rentang: `True` atau `False` Jenis nilai: Boolean HPO dapat disetel: Ya
Hiperparameter featurisasi	

Nama	Penjelasan
`min_user_history_length_percentile`	Persentil minimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `min_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang riwayat pendek. Pengguna dengan riwayat singkat sering menunjukkan pola berdasarkan popularitas item, bukan kebutuhan atau keinginan pribadi pengguna. Menghapusnya dapat melatih model dengan lebih fokus pada pola yang mendasarinya dalam data Anda. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna, menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menetapkan nilai yang mempertahankan mayoritas pengguna, tetapi menghapus kasus tepi. Misalnya, menyetel `min__user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0.0 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`max_user_history_length_percentile`	Persentil maksimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `max_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang sejarah panjang karena data untuk pengguna ini cenderung mengandung noise. Misalnya, robot mungkin memiliki daftar panjang interaksi otomatis. Menghapus pengguna ini membatasi kebisingan dalam pelatihan. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan mayoritas pengguna tetapi menghapus kasus tepi. Misalnya, menyetel `min__user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0,99 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Resep HRNN-metadata (warisan)

> **Note**
>
> Resep HRNN Legacy tidak lagi tersedia. Dokumentasi ini untuk tujuan referensi.
> Kami merekomendasikan penggunaan resep aws-user-personalizaton (User-Personalization) di atas resep HRNN warisan. Personalisasi Pengguna meningkatkan dan menyatukan

fungsionalitas yang ditawarkan oleh resep HRNN. Untuk informasi selengkapnya, lihat Resep Personalisasi Pengguna.

Resep HRNN-metadata memprediksi item yang akan berinteraksi dengan pengguna. Ini mirip dengan HRNN resep, dengan fitur tambahan yang berasal dari metadata kontekstual, pengguna, dan item (masing-masing dari kumpulan data Interaksi, Pengguna, dan Item). HRNN-metadata memberikan manfaat akurasi dibandingkan model non-metadata ketika metadata berkualitas tinggi tersedia. Menggunakan resep ini mungkin membutuhkan waktu pelatihan yang lebih lama.

Resep HRNN-metadata memiliki sifat-sifat berikut:

- Nama – `aws-hrnn-metadata`
- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-hrnn-metadata`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-hrnn-metadata`
- Transformasi fitur ARN — `arn:aws:personalize:::feature-transformation/featurize_metadata`
- Jenis resep — `USER_PERSONALIZATION`

Tabel berikut menjelaskan hiperparameter untuk resep HRNN-metadata. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Hyperparameter featurisasi mengontrol cara memfilter data yang akan digunakan dalam pelatihan. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Dapatkah parameter berpartisipasi dalam optimasi hyperparameter (HPO)?

Nama	Penjelasan
Algoritma Hyperparameter	

Nama	Penjelasan
`hidden_dimension`	Jumlah variabel tersembunyi yang digunakan dalam model. Variabel tersembunyi membuat ulang riwayat pembelian pengguna dan statistik item untuk menghasilkan skor peringkat. Tentukan lebih banyak dimensi tersembunyi saat kumpulan data interaksi Item Anda menyertakan pola yang lebih rumit. Menggunakan dimensi yang lebih tersembunyi membutuhkan dataset yang lebih besar dan lebih banyak waktu untuk diproses. Untuk memutuskan nilai optimal, gunakan HPO. Untuk menggunakan HPO, atur `performHPO` ke `true` saat Anda memanggil CreateSolution dan CreateSolutionVersion operasi. Nilai default: 43 Rentang: [32, 256] Jenis nilai: Integer HPO dapat disetel: Ya

Nama	Penjelasan
`bptt`	Menentukan apakah akan menggunakan teknik propagasi balik melalui waktu. Propagasi balik melalui waktu adalah teknik yang memperbarui bobot dalam algoritme berbasis jaringan saraf berulang. Gunakan kredit jangka panjang `bptt` untuk menghubungkan hadiah yang tertunda ke acara awal. Misalnya, hadiah yang tertunda dapat berupa pembelian yang dilakukan setelah beberapa klik. Acara awal bisa menjadi klik awal. Bahkan dalam jenis acara yang sama, seperti klik, ada baiknya untuk mempertimbangkan efek jangka panjang dan memaksimalkan total hadiah. Untuk mempertimbangkan efek jangka panjang, gunakan `bptt` nilai yang lebih besar. Menggunakan `bptt` nilai yang lebih besar membutuhkan kumpulan data yang lebih besar dan lebih banyak waktu untuk diproses. Nilai default: 32 Rentang: [2, 32] Jenis nilai: Integer HPO dapat disetel: Ya

Nama	Penjelasan
`recency_mask`	Menentukan apakah model harus mempertimbangkan tren popularitas terbaru dalam kumpulan data interaksi Item. Tren popularitas terbaru mungkin termasuk perubahan mendadak dalam pola yang mendasari peristiwa interaksi. Untuk melatih model yang memberi bobot lebih pada peristiwa baru-baru ini, atur `recency_mask` ke`true`. Untuk melatih model yang sama-sama menimbang semua interaksi masa lalu, atur `recency_mask` ke`false`. Untuk mendapatkan rekomendasi yang baik menggunakan bobot yang sama, Anda mungkin memerlukan kumpulan data pelatihan yang lebih besar. Nilai default: `True` Rentang: `True` atau `False` Jenis nilai: Boolean HPO dapat disetel: Ya
Hiperparameter featurisasi	

Nama	Penjelasan
`min_user_history_length_percentile`	Persentil minimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `min_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang riwayat pendek. Pengguna dengan riwayat singkat sering menunjukkan pola berdasarkan popularitas item, bukan kebutuhan atau keinginan pribadi pengguna. Menghapusnya dapat melatih model dengan lebih fokus pada pola yang mendasarinya dalam data Anda. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna, menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menetapkan nilai yang mempertahankan mayoritas pengguna, tetapi menghapus kasus tepi. Misalnya, menyetel `min__user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0.0 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`max_user_history_length_percentile`	Persentil maksimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `max_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang sejarah panjang karena data untuk pengguna ini cenderung mengandung noise. Misalnya, robot mungkin memiliki daftar panjang interaksi otomatis. Menghapus pengguna ini membatasi kebisingan dalam pelatihan. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan mayoritas pengguna tetapi menghapus kasus tepi. Misalnya, menyetel `min__user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0,99 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Resep HRNN-ColdStart (warisan)

> **Note**
>
> Resep HRNN Legacy tidak lagi tersedia. Dokumentasi ini untuk tujuan referensi.
> Kami merekomendasikan penggunaan resep aws-user-personalizaton (User-Personalization) di atas resep HRNN warisan. Personalisasi Pengguna meningkatkan dan menyatukan

fungsionalitas yang ditawarkan oleh resep HRNN. Untuk informasi selengkapnya, lihat Resep Personalisasi Pengguna.

Gunakan resep HRNN-ColdStart untuk memprediksi item yang akan berinteraksi dengan pengguna saat Anda sering menambahkan item dan interaksi baru dan ingin segera mendapatkan rekomendasi untuk item tersebut. Resep HRNN-ColdStart mirip dengan Metadata HRNN resepnya, tetapi memungkinkan Anda mendapatkan rekomendasi untuk item baru.

Selain itu, Anda dapat menggunakan resep HRNNN-ColdStart ketika Anda ingin mengecualikan dari item pelatihan yang memiliki daftar panjang interaksi baik karena tren popularitas baru-baru ini atau karena interaksi mungkin sangat tidak biasa dan menimbulkan kebisingan dalam pelatihan. Dengan HRNN-ColdStart, Anda dapat memfilter item yang kurang relevan untuk membuat subset untuk pelatihan. Subset item, yang disebut item dingin, adalah item yang memiliki peristiwa interaksi terkait dalam kumpulan data interaksi Item. Item dianggap sebagai item dingin jika memiliki yang berikut:

- Interaksi lebih sedikit daripada jumlah interaksi maksimum yang ditentukan. Anda menentukan nilai ini dalam `cold_start_max_interactions` hyperparameter resep.
- Durasi relatif yang lebih pendek dari durasi maksimum. Anda menentukan nilai ini dalam `cold_start_max_duration` hyperparameter resep.

Untuk mengurangi jumlah item dingin, tetapkan nilai yang lebih rendah untuk `cold_start_max_interactions` atau`cold_start_max_duration`. Untuk menambah jumlah item dingin, tetapkan nilai yang lebih besar untuk `cold_start_max_interactions` atau`cold_start_max_duration`.

HRNN-ColdStart memiliki batas item dingin berikut:

- `Maximum cold start items`: 80.000
- `Minimum cold start items`: 100

Jika jumlah item dingin berada di luar kisaran ini, upaya untuk membuat solusi akan gagal.

Resep HRNN-ColdStart memiliki sifat-sifat berikut:

- Nama – `aws-hrnn-coldstart`

- Resep Nama Sumber Daya Amazon (ARN) — `arn:aws:personalize:::recipe/aws-hrnn-coldstart`
- Algoritma ARN — `arn:aws:personalize:::algorithm/aws-hrnn-coldstart`
- Transformasi fitur ARN — `arn:aws:personalize:::feature-transformation/featurize_coldstart`
- Jenis resep — `USER_PERSONALIZATION`

Untuk informasi selengkapnya, lihat Memilih resep.

Tabel berikut menjelaskan hyperparameters untuk resep HRNN-ColdStart. Hyperparameter adalah parameter algoritma yang dapat Anda sesuaikan untuk meningkatkan kinerja model. Algoritma hyperparameters mengontrol bagaimana kinerja model. Hyperparameter featurisasi mengontrol cara memfilter data yang akan digunakan dalam pelatihan. Proses memilih nilai terbaik untuk hyperparameter disebut optimasi hyperparameter (HPO). Untuk informasi selengkapnya, lihat Hyperparameters dan HPO.

Tabel ini juga menyediakan informasi berikut untuk setiap hyperparameter:

- Rentang: [batas bawah, batas atas]
- Jenis nilai: Integer, Continuous (float), Kategoris (Boolean, daftar, string)
- HPO tunable: Dapatkah parameter berpartisipasi dalam HPO?

Nama	Penjelasan
Hiperparameter algoritma	
`hidden_dimension`	Jumlah variabel tersembunyi yang digunakan dalam model. Variabel tersembunyi membuat ulang riwayat pembelian pengguna dan statistik item untuk menghasilkan skor peringkat. Tentukan lebih banyak dimensi tersembunyi saat kumpulan data interaksi Item Anda menyertakan pola yang lebih rumit. Menggunakan dimensi yang lebih tersembunyi membutuhkan dataset yang lebih besar dan lebih banyak waktu untuk diproses. Untuk memutuskan nilai optimal, gunakan HPO. Untuk menggunakan HPO, atur `performHPO`

Nama	Penjelasan
	ke `true` saat Anda memanggil CreateSolution dan CreateSolutionVersion operasi. Nilai default: 149 Rentang: [32, 256] Jenis nilai: Integer HPO dapat disetel: Ya
`bptt`	Menentukan apakah akan menggunakan teknik propagasi balik melalui waktu. Propagasi balik melalui waktu adalah teknik yang memperbarui bobot dalam algoritme berbasis jaringan saraf berulang. Gunakan kredit jangka panjang `bptt` untuk menghubungkan hadiah yang tertunda ke acara awal. Misalnya, hadiah yang tertunda dapat berupa pembelian yang dilakukan setelah beberapa klik. Acara awal bisa menjadi klik awal. Bahkan dalam jenis acara yang sama, seperti klik, ada baiknya untuk mempertimbangkan efek jangka panjang dan memaksimalkan total hadiah. Untuk mempertimbangkan efek jangka panjang, gunakan `bptt` nilai yang lebih besar. Menggunakan `bptt` nilai yang lebih besar membutuhkan kumpulan data yang lebih besar dan lebih banyak waktu untuk diproses. Nilai default: 32 Rentang: [2, 32] Jenis nilai: Integer HPO dapat disetel: Ya

Nama	Penjelasan
`recency_mask`	Menentukan apakah model harus mempertimbangkan tren popularitas terbaru dalam kumpulan data interaksi Item. Tren popularitas terbaru mungkin termasuk perubahan mendadak dalam pola yang mendasari peristiwa interaksi. Untuk melatih model yang memberi bobot lebih pada peristiwa baru-baru ini, atur `recency_mask` ke `true`. Untuk melatih model yang sama-sama menimbang semua interaksi masa lalu, atur `recency_mask` ke `false`. Untuk mendapatkan rekomendasi yang baik menggunakan bobot yang sama, Anda mungkin memerlukan kumpulan data pelatihan yang lebih besar. Nilai default: `True` Rentang: `True` atau `False` Jenis nilai: Boolean HPO dapat disetel: Ya
Hiperparameter featurisasi	
`cold_start_max_interactions`	Jumlah maksimum interaksi item-pengguna suatu item harus dianggap sebagai item dingin. Nilai default: 15 Rentang: Bilangan bulat positif Jenis nilai: Integer HPO dapat disetel: Tidak

Nama	Penjelasan
`cold_start_max_duration`	Durasi maksimum dalam beberapa hari relatif terhadap titik awal untuk interaksi item pengguna dianggap sebagai item awal yang dingin. Untuk mengatur titik awal interaksi user-item, atur hyperparameter. `cold_start_relative_from` Nilai default: 5.0 Rentang: Pelampung positif Jenis nilai: Float HPO dapat disetel: Tidak
`cold_start_relative_from`	Menentukan titik awal untuk resep HRNN-ColdStart untuk dihitung. `cold_start_max_duration` Untuk menghitung dari waktu saat ini, pilih`currentTime` . Untuk menghitung `cold_start_max_duration` dari stempel waktu item terbaru di Kumpulan data interaksi item, pilih. `latestItem` Pengaturan ini berguna jika Anda sering menambahkan item baru. Nilai default: `latestItem` Rentang:`currentTime` , `latestItem` Jenis nilai: String HPO dapat disetel: Tidak

Nama	Penjelasan
`min_user_history_length_percentile`	Persentil minimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `min_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang riwayat pendek. Pengguna dengan riwayat singkat sering menunjukkan pola berdasarkan popularitas item, bukan kebutuhan atau keinginan pribadi pengguna. Menghapusnya dapat melatih model dengan lebih fokus pada pola yang mendasarinya dalam data Anda. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna, menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan mayoritas pengguna, tetapi menghapus kasus tepi. Misalnya, menyetel `min__user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0.0 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Nama	Penjelasan
`max_user_history_length_percentile`	Persentil maksimum panjang riwayat pengguna untuk dimasukkan dalam pelatihan model. Panjang riwayat adalah jumlah total data tentang pengguna. Gunakan `max_user_history_length_percentile` untuk mengecualikan persentase pengguna dengan panjang sejarah panjang karena data untuk pengguna ini cenderung mengandung noise. Misalnya, robot mungkin memiliki daftar panjang interaksi otomatis. Menghapus pengguna ini membatasi kebisingan dalam pelatihan. Pilih nilai yang sesuai setelah Anda meninjau panjang riwayat pengguna menggunakan histogram atau alat serupa. Kami merekomendasikan untuk menyetel nilai yang mempertahankan mayoritas pengguna tetapi menghapus kasus tepi. Misalnya, menyetel `min__user_history_length_percentile to 0.05` dan `max_user_history_length_percentile to 0.95` menyertakan semua pengguna kecuali yang memiliki panjang riwayat di bagian bawah atau atas 5%. Nilai default: 0,99 Rentang: [0.0, 1.0] Jenis nilai: Float HPO dapat disetel: Tidak

Mempersiapkan data pelatihan untuk Amazon Personalize

Setelah Anda memilih kasus penggunaan domain atau resep dan mencatat persyaratan datanya, Anda siap untuk mulai menyiapkan data Anda. Amazon Personalize dapat menggunakan jenis data berikut:

- Interaksi item — Di Amazon Personalize, interaksi item adalah peristiwa interaksi positif antara pengguna dan item di katalog Anda. Misalnya, pengguna menonton film, melihat daftar, atau membeli sepasang sepatu.
- Item — Metadata item mungkin mencakup informasi seperti harga, jenis SKU, deskripsi, atau ketersediaan untuk setiap item dalam katalog Anda.
- Pengguna — Metadata pengguna dapat mencakup informasi seperti usia, jenis kelamin, keanggotaan loyalitas, dan minat untuk setiap pengguna Anda.
- Tindakan — Tindakan adalah aktivitas keterlibatan yang mungkin ingin Anda rekomendasikan kepada pelanggan Anda. Tindakan mungkin termasuk menginstal aplikasi seluler Anda, melengkapi profil keanggotaan, bergabung dengan program loyalitas Anda, atau mendaftar untuk email promosi. Untuk Next-Best-Action resepnya, kumpulan data Tindakan diperlukan. Tidak ada resep kustom atau kasus penggunaan domain lain yang menggunakan data Tindakan.
- Interaksi aksi — Interaksi aksi adalah peristiwa interaksi antara pengguna dan tindakan. Next-Best-ActionResepnya menggunakan data ini dan data dalam kumpulan data Tindakan Anda untuk merekomendasikan tindakan kepada pengguna Anda. Tidak ada resep khusus atau kasus penggunaan domain lain yang menggunakan data Interaksi Tindakan.

Amazon Personalize menyimpan data dalam kumpulan data, satu untuk setiap jenis data. Setiap dataset memiliki persyaratan yang berbeda. Saat mengimpor data ke kumpulan data Amazon Personalize, Anda dapat memilih untuk mengimpor rekaman secara massal, satu per satu, atau keduanya. Impor massal melibatkan pengimporan sejumlah besar catatan historis yang disimpan dalam satu atau beberapa file CSV di bucket Amazon S3.

- Jika tidak memiliki data massal, Anda dapat menggunakan operasi impor individual untuk mengumpulkan data dan streaming peristiwa hingga memenuhi persyaratan pelatihan Amazon Personalize dan persyaratan data kasus atau resep penggunaan domain Anda. Untuk informasi tentang merekam acara, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi. Untuk informasi tentang mengimpor catatan individual, lihatMengimpor catatan individual ke dalam kumpulan data Amazon Personalize.

- Jika Anda tidak yakin memiliki cukup data atau jika Anda memiliki pertanyaan tentang kualitasnya, Anda dapat mengimpor data Anda ke kumpulan data Amazon Personalize dan menggunakan Amazon Personalize untuk menganalisisnya. Untuk informasi selengkapnya, lihat Menganalisis kualitas dan kuantitas data di Amazon Personalisasi kumpulan data.

Bagian berikut menyediakan persyaratan data untuk setiap jenis kumpulan data Amazon Personalize dan pedoman untuk menyiapkan data massal. Jika Anda tidak memiliki data massal, tinjau bagian untuk memahami data yang diperlukan dan opsional yang dapat Anda impor dengan operasi impor individual. Jika Anda memerlukan bantuan tambahan untuk memformat data, Anda dapat menggunakan Amazon SageMaker AI Data Wrangler (Data Wrangler) untuk menyiapkan data Anda. Untuk informasi selengkapnya, lihat Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler.

Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

Topik

- Pedoman format data massal untuk semua jenis data
- Mempersiapkan data interaksi item untuk pelatihan
- Mempersiapkan metadata item untuk pelatihan
- Mempersiapkan metadata pengguna untuk pelatihan
- Mempersiapkan metadata tindakan untuk pelatihan
- Mempersiapkan data interaksi tindakan untuk pelatihan

Pedoman format data massal untuk semua jenis data

Panduan dan persyaratan berikut dapat membantu Anda memastikan data massal Anda diformat dengan benar.

- Data masukan Anda harus dalam file CSV (nilai yang dipisahkan koma).
- Baris pertama file CSV Anda harus berisi header kolom Anda. Jangan lampirkan header dalam tanda kutip (“).
- Kolom harus memiliki nama alfanumerik yang unik. Misalnya, Anda tidak dapat menambahkan `GENRES_FIELD_1` bidang dan `GENRESFIELD1` bidang.

- Jika Anda mengimpor beberapa file CSV, semua header kolom harus cocok di semua file.
- Pastikan Anda memiliki bidang yang diperlukan untuk jenis kumpulan data Anda dan pastikan namanya selaras dengan persyaratan Amazon Personalize. Misalnya, data Item Anda mungkin memiliki kolom `ITEM_IDENTIFICATION_NUMBER` yang dipanggil IDs untuk setiap item Anda. Untuk menggunakan kolom ini sebagai bidang ITEM_ID, ganti nama kolom menjadi. `ITEM_ID` Jika Anda menggunakan Data Wrangler untuk memformat data, Anda dapat menggunakan kolom Peta untuk transformasi Amazon Personalize Data Wrangler untuk memastikan kolom Anda diberi nama dengan benar.

  Untuk informasi tentang menggunakan Data Wrangler untuk menyiapkan data Anda, lihat. Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler
- Setiap catatan dalam file CSV Anda harus dalam satu baris.
- Amazon Personalize tidak mendukung tipe data yang kompleks seperti array dan peta.
- Agar Amazon Personalisasi menggunakan data boolean saat melatih atau memfilter, gunakan nilai string `"True"` dan `"False"` atau nilai numerik `1` untuk true dan false. `0`
- Jika Anda menggunakan Data Wrangler untuk memformat data Anda, Anda dapat menggunakan Data Wrangler mengubah Nilai Parse sebagai Jenis untuk mengonversi tipe data.
- `TIMESTAMP`dan `CREATION_TIMESTAMP` data harus dalam format waktu epoch UNIX. Untuk informasi selengkapnya, lihat Data stempel waktu.
- Hindari menyertakan '' karakter atau karakter khusus dalam ID item, ID pengguna, dan data ID tindakan.
- Jika data Anda menyertakan karakter yang tidak dikodekan ASCII, file CSV Anda harus dikodekan dalam format UTF-8.
- Pastikan Anda memformat data tekstual apa pun seperti yang dijelaskan dalamMetadata teks tidak terstruktur.

Mempersiapkan data interaksi item untuk pelatihan

Interaksi item adalah peristiwa interaksi positif antara pengguna dan item dalam katalog Anda. Misalnya, pengguna menonton film, melihat daftar, atau membeli sepasang sepatu. Anda mengimpor data tentang interaksi pengguna dengan item Anda ke dalam kumpulan data interaksi Item. Anda dapat merekam beberapa jenis acara, seperti klik, tonton, atau pembelian.

Misalnya, jika pengguna mengklik item tertentu dan kemudian menyukai item tersebut, Anda dapat meminta Amazon Personalisasi menggunakan peristiwa ini sebagai data pelatihan. Untuk setiap

peristiwa, Anda akan merekam ID pengguna, ID item, stempel waktu (dalam format zaman waktu Unix), dan jenis acara (klik dan suka). Anda kemudian akan menambahkan kedua peristiwa interaksi item ke kumpulan data interaksi Item.

Untuk semua kasus penggunaan domain dan resep khusus, data interaksi item massal Anda harus dalam file CSV. Setiap baris harus mewakili interaksi tunggal antara pengguna dan item. Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

Bagian berikut memberikan informasi selengkapnya tentang cara menyiapkan data interaksi item Anda untuk Amazon Personalize. Untuk pedoman format data massal untuk semua jenis data, lihat pedoman format data massal

Topik

- Persyaratan data interaksi item
- Data stempel waktu
- Jenis peristiwa dan data nilai acara
- Metadata kontekstual
- Data tayangan
- Contoh data interaksi

Persyaratan data interaksi item

Bagian berikut mencantumkan persyaratan data interaksi item untuk Amazon Personalize. Untuk kuota tambahan, lihatTitik akhir dan kuota Amazon Personalize.

Persyaratan pelatihan minimum

Untuk semua kasus penggunaan domain dan resep khusus, data interaksi item massal Anda harus memiliki yang berikut:

- Minimal 1000 catatan interaksi item dari pengguna yang berinteraksi dengan item di katalog Anda. Interaksi ini dapat berasal dari impor massal, atau acara streaming, atau keduanya.
- Minimal 25 pengguna unik IDs dengan setidaknya dua interaksi item untuk masing-masing.

Untuk rekomendasi kualitas, kami sarankan Anda memiliki setidaknya 50.000 interaksi item dari setidaknya 1.000 pengguna dengan dua atau lebih interaksi item masing-masing.

Untuk membuat pemberi rekomendasi atau solusi khusus, Anda harus setidaknya membuat kumpulan data interaksi Item.

Persyaratan kolom

Data interaksi item Anda harus memiliki kolom berikut.

- USER_ID — Pengidentifikasi unik dari pengguna yang berinteraksi dengan item. Setiap acara harus memiliki USER_ID. Itu harus `string` dengan panjang maksimal 256 karakter.
- ITEM_ID — Pengidentifikasi unik dari item yang berinteraksi dengan pengguna. Setiap acara harus memiliki ID item. Itu harus `string` dengan panjang maksimal 256 karakter.
- TIMESTAMP — Waktu kejadian terjadi (dalam format waktu epoch Unix dalam hitungan detik). Setiap interaksi harus memiliki TIMESTAMP. Untuk informasi selengkapnya, lihat Data stempel waktu.
- EVENT_TYPE - Sifat peristiwa interaksi item, seperti klik, tonton, atau pembelian. Untuk pemberi rekomendasi domain, Anda harus memiliki kolom tipe acara dan setiap interaksi harus memiliki jenis acara. Untuk semua resep khusus, kolom EVENT_TYPE direkomendasikan tetapi opsional. Jika Anda menambahkannya, setiap acara harus memiliki jenis acara. Untuk mengetahui informasi selengkapnya, lihat Jenis peristiwa dan data nilai acara.

Anda bebas menambahkan kolom kustom tambahan tergantung pada kasus penggunaan dan data Anda. Jumlah maksimum kolom metadata opsional adalah 5. Kolom ini dapat mencakup empty/null nilai. Kami menyarankan agar kolom-kolom ini minimal 70 persen selesai.

Data stempel waktu

Data stempel waktu harus dalam format waktu epoch Unix dalam hitungan detik. Misalnya, stempel waktu Epoch dalam detik untuk tanggal 31 Juli 2020 adalah 1596238243. Untuk mengonversi tanggal ke stempel waktu epoch Unix, gunakan konverter Epoch - konverter stempel waktu Unix.

Amazon Personalize menggunakan data stempel waktu untuk menghitung kebaruan dan mengidentifikasi pola berbasis waktu. Ini membantu Amazon Personalisasi menjaga rekomendasi up-to-date dengan preferensi pengguna yang terus berkembang.

Jenis peristiwa dan data nilai acara

Kumpulan data interaksi Item dapat menyimpan jenis peristiwa dan data nilai peristiwa untuk setiap interaksi. Hanya sumber daya khusus yang menggunakan data nilai peristiwa.

Data tipe acara

Jenis peristiwa interaksi item memberikan konteks tentang sifat dan signifikansinya. Contoh jenis acara mungkin klik, tonton, atau beli. Amazon Personalize menggunakan data tipe peristiwa, seperti data klik atau beli, untuk mengidentifikasi maksud dan minat pengguna. Jumlah maksimum jenis peristiwa berbeda yang dikombinasikan dengan jumlah total kolom metadata opsional dalam kumpulan data interaksi Item adalah 10.

Untuk pemberi rekomendasi domain, Anda harus memiliki kolom tipe acara dan setiap interaksi harus memiliki jenis acara. Untuk semua resep khusus, kolom EVENT_TYPE direkomendasikan tetapi opsional. Jika Anda menambahkannya, setiap acara harus memiliki jenis acara.

Jika Anda membuat sumber daya khusus, Anda dapat memilih peristiwa yang digunakan untuk pelatihan berdasarkan jenis acara.Jika kumpulan data Anda memiliki beberapa jenis peristiwa di kolom EVENT_TYPE, dan Anda tidak memberikan jenis peristiwa saat mengonfigurasi solusi khusus, Amazon Personalize menggunakan semua data interaksi item untuk pelatihan dengan bobot yang sama terlepas dari jenisnya. Untuk informasi selengkapnya, lihat Memilih data interaksi item yang digunakan untuk pelatihan.

Jika Anda memiliki beberapa jenis acara dan menggunakan resep User-Personalization-v 2 atau resep Personalized-Ranking-v 2, saat Anda mengonfigurasi solusi khusus, Anda dapat menentukan bobot yang berbeda untuk jenis yang berbeda. Misalnya, Anda dapat mengonfigurasi solusi untuk memberikan bobot lebih untuk membeli acara daripada peristiwa klik. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi dengan konfigurasi peristiwa.

Kasus penggunaan berikut memiliki persyaratan jenis acara tertentu:

Kasus penggunaan domain VIDEO_ON_DEMAND

- Karena Anda menonton X membutuhkan minimal 1000 `Watch` acara.
- Paling populer membutuhkan minimal 1000 `Watch` acara.

Kasus penggunaan domain ECOMMERCE

- Paling banyak dilihat membutuhkan minimal 1000 `View` acara.
- Penjual terbaik membutuhkan minimal 1000 `Purchase` acara.

Jenis peristiwa positif dan negatif

Amazon Personalize mengasumsikan interaksi apa pun adalah interaksi positif. Interaksi dengan jenis peristiwa negatif, seperti tidak suka, tidak akan selalu mencegah item muncul di rekomendasi future pengguna.

Berikut ini adalah cara untuk memiliki peristiwa negatif dan rekomendasi pengaruh ketidaktertarikan pengguna:

- Untuk semua kasus penggunaan domain dan Personalisasi Pengguna resepnya, Amazon Personalize dapat menggunakan data tayangan. Ketika item muncul di data tayangan dan pengguna tidak memilihnya, item tersebut cenderung tidak muncul dalam rekomendasi. Untuk informasi selengkapnya, lihat Data tayangan.
- Jika Anda menggunakan sumber daya khusus dan mengimpor jenis peristiwa positif dan negatif, Anda dapat melatih hanya pada jenis peristiwa positif dan kemudian menyaring item yang berinteraksi dengan pengguna secara negatif. Untuk informasi selengkapnya, lihat Memilih data interaksi item yang digunakan untuk pelatihan dan Memfilter rekomendasi dan segmen pengguna.

Data nilai acara (sumber daya khusus)

Data nilai peristiwa mungkin persentase film yang ditonton pengguna atau peringkat dari 10. Jika Anda membuat solusi khusus, Anda dapat memilih catatan yang digunakan untuk pelatihan berdasarkan data di kolom EVENT_TYPE dan EVENT_VALUE. Dengan pemberi rekomendasi domain, Amazon Personalize tidak menggunakan data nilai peristiwa dan Anda tidak dapat memfilter peristiwa sebelum pelatihan.

Untuk memilih rekaman berdasarkan jenis dan nilai, rekam jenis peristiwa dan data nilai peristiwa untuk peristiwa. Tidak semua acara harus memiliki nilai acara. Nilai yang Anda pilih untuk setiap acara tergantung pada data apa yang ingin Anda kecualikan dan jenis acara apa yang Anda rekam. Misalnya, Anda mungkin mencocokkan aktivitas pengguna, seperti persentase video yang ditonton pengguna untuk jenis acara tontonan.

Saat mengonfigurasi solusi, Anda menetapkan nilai tertentu sebagai ambang batas untuk mengecualikan catatan dari pelatihan. Misalnya, jika data EVENT_VALUE untuk peristiwa dengan

EVENT_TYPE jam tangan adalah persentase video yang ditonton pengguna, jika Anda menetapkan ambang nilai peristiwa ke 0,5, dan jenis acara yang akan ditonton, Amazon Personalize melatih model hanya menggunakan menonton peristiwa interaksi dengan EVENT_VALUE lebih besar dari atau sama dengan 0,5.

Untuk informasi selengkapnya, lihat Memilih data interaksi item yang digunakan untuk pelatihan

Metadata kontekstual

Dengan resep dan kasus penggunaan pemberi rekomendasi tertentu, Amazon Personalize dapat menggunakan metadata kontekstual saat mengidentifikasi pola dasar yang mengungkapkan item yang paling relevan untuk pengguna Anda. Metadata kontekstual adalah data interaksi yang Anda kumpulkan di lingkungan pengguna pada saat kejadian, seperti lokasi atau jenis perangkat mereka. Anda juga dapat menentukan konteks pengguna ketika Anda mendapatkan rekomendasi untuk pengguna.

Sertakan metadata kontekstual untuk memberikan pengalaman yang lebih personal bagi pengguna Anda dan kurangi fase awal dingin untuk pengguna baru. Fase cold-start adalah ketika rekomendasi kurang relevan karena kurangnya data pengguna historis.

Misalnya, jika file CSV interaksi item Anda menyertakan kolom DEVICE_TYPE dengan dan nilai`tablet`, `phone` Amazon Personalize dapat mempelajari cara pelanggan berbelanja secara berbeda dengan perangkat yang berbeda. Ketika Anda mendapatkan rekomendasi untuk pengguna, Anda dapat menentukan perangkat mereka dan rekomendasi akan lebih relevan, bahkan jika pengguna tidak memiliki riwayat interaksi.

Berikut ini menunjukkan bagaimana Anda akan memformat file CSV interaksi item dengan kolom DEVICE_TYPE sebagai metadata kontekstual.

```
ITEM_ID,USER_ID,TIMESTAMP,DEVICE_TYPE,EVENT_TYPE
shoe12345,12,1428624000,Tablet,CLICK
shoe12346,12,1420416000,Tablet,CLICK
shoe12347,12,1410652800,Tablet,BUY
shoe4444,13,1409961600,Phone,CLICK
shoe4445,13,1402876800,Phone,BUY
shoe4336,13,1402185600,Phone,CLICK
.....
```

Untuk grup kumpulan data Domain, kasus penggunaan pemberi rekomendasi berikut dapat menggunakan metadata kontekstual:

- Direkomendasikan untuk Anda(Domain eCommerce)
- Pilihan teratas untuk Anda(Domain VIDEO_ON_DEMAND)

Untuk sumber daya kustom, resep yang menggunakan metadata kontekstual mencakup yang berikut:

- Personalisasi pengguna-v2 dan Personalisasi Pengguna
- Personalisasi-peringkat-v2 dan Peringkat yang Dipersonalisasi

Untuk informasi tentang menyertakan konteks saat Anda mendapatkan rekomendasi, lihatMeningkatkan relevansi rekomendasi dengan metadata kontekstual. Untuk contoh ujung ke ujung yang menunjukkan cara menggunakan metadata kontekstual, lihat posting Blog AWS Machine Learning berikut: Meningkatkan relevansi rekomendasi Amazon Personalize Anda dengan memanfaatkan informasi kontekstual.

Data tayangan

Tayangan adalah daftar item yang terlihat oleh pengguna saat berinteraksi dengan (misalnya, diklik atau ditonton) item tertentu. Jika Anda menggunakan kasus penggunaan domain yang menyediakan personalisasi atau Personalisasi Pengguna resep, Amazon Personalize dapat menggunakan data tayangan untuk memandu eksplorasi.

Dengan eksplorasi, rekomendasi mencakup beberapa item atau tindakan yang biasanya cenderung tidak direkomendasikan bagi pengguna, seperti item atau tindakan baru, item atau tindakan dengan sedikit interaksi, atau item atau tindakan yang kurang relevan bagi pengguna berdasarkan perilaku mereka sebelumnya. Semakin sering suatu item muncul dalam data tayangan, semakin kecil kemungkinan Amazon Personalize menyertakan item dalam eksplorasi.

Saat Anda membuat pemberi rekomendasi atau solusi, Amazon Personalize selalu mengecualikan data tayangan dari pelatihan. Ini karena Amazon Personalize tidak melatih model Anda dengan data tayangan. Sebaliknya, ia menggunakannya ketika Anda mendapatkan rekomendasi untuk memandu eksplorasi bagi pengguna.

Nilai tayangan dapat memiliki paling banyak 1000 karakter (termasuk karakter batang vertikal). Untuk grup kumpulan data Domain, kasus penggunaan pemberi rekomendasi berikut dapat menggunakan data tayangan:

- Direkomendasikan untuk Anda(Domain eCommerce)
- Pilihan teratas untuk Anda(Domain VIDEO_ON_DEMAND)

Untuk informasi lebih lanjut tentang eksplorasi lihatEksplorasi. Amazon Personalize dapat memodelkan dua jenis tayangan: Tayangan implisit dan. Tayangan eksplisit

Tayangan eksplisit

Tayangan eksplisit adalah tayangan yang Anda rekam dan kirim secara manual ke Amazon Personalisasi. Gunakan tayangan eksplisit untuk memanipulasi hasil dari Amazon Personalize. Urutan item tidak berdampak.

Misalnya, Anda mungkin memiliki aplikasi belanja yang memberikan rekomendasi untuk sepatu. Jika Anda hanya merekomendasikan sepatu yang saat ini tersedia, Anda dapat menentukan item ini menggunakan tayangan eksplisit. Alur kerja rekomendasi Anda menggunakan tayangan eksplisit mungkin sebagai berikut:

1. Anda meminta rekomendasi untuk salah satu pengguna menggunakan Amazon Personalize APIthe section called “GetRecommendations”.
2. Amazon Personalize menghasilkan rekomendasi untuk pengguna yang menggunakan model Anda (versi solusi) dan mengembalikannya dalam respons API.
3. Anda hanya menunjukkan kepada pengguna sepatu yang direkomendasikan yang tersedia.
4. Untuk impor data inkremental real-time, ketika pengguna berinteraksi dengan (misalnya, mengklik) sepasang sepatu, Anda mencatat pilihan dalam panggilan ke PutEvents API dan mencantumkan item yang direkomendasikan yang tersedia dalam parameter. `impression` Untuk contoh kode lihatMerekam peristiwa interaksi item dengan data tayangan.

   Untuk mengimpor tayangan dalam data interaksi item historis, Anda dapat mencantumkan tayangan eksplisit di file csv Anda dan memisahkan setiap item dengan karakter '| '. Karakter batang vertikal dihitung menuju batas 1000 karakter. Sebagai contoh lihatMemformat tayangan eksplisit.
5. Amazon Personalize menggunakan data tayangan untuk memandu eksplorasi, di mana rekomendasi future menyertakan sepatu baru dengan data interaksi atau relevansi yang lebih sedikit.

Memformat tayangan eksplisit

Untuk menyertakan tayangan eksplisit dalam file CSV Anda, tambahkan kolom IMPRESSION. Untuk setiap interaksi item, tambahkan daftar ItemID yang dipisahkan dengan bilah vertikal, '|', karakter. Karakter batang vertikal dihitung menuju batas 1000 karakter untuk data tayangan. Jika Anda menyertakan tayangan eksplisit dalam PutEvents operasi, Anda menentukan item dalam array string.

Berikut ini adalah kutipan singkat dari file CSV yang menyertakan tayangan eksplisit di kolom. `IMPRESSION`

EVENT_TYPE	KESAN	ITEM_ID	TIMESTAMP	USER_ID
klik	73|70|17 |95|96	73	1586731606	PENGGUNA_1
klik	35 | 82 | 78 | 57 | 20 | 63 | 1 | 90 | 76 | 75 | 49 | 71 | 26 | 24 | 25 | 6	35	1586735164	PENGGUNA_2
...	...	...	...	...

Aplikasi menunjukkan `USER_1` item pengguna`73`,`70`, `1795`,, `96` dan pengguna akhirnya memilih item`73`. Saat Anda membuat versi solusi baru berdasarkan data ini, item`70`,`17`,`95`, dan `96` akan lebih jarang direkomendasikan kepada penggunaUSER_1.

Tayangan implisit

Tayangan implisit adalah rekomendasi, diambil dari Amazon Personalize, yang Anda tunjukkan kepada pengguna. File CSV Anda tidak perlu menyertakan kolom IMPRESSION atau RECOMMENTATION_ID untuk menggunakan tayangan implisit. Sebagai gantinya, Anda menyertakan `RecommendationId` (dikembalikan oleh GetRecommendations dan GetPersonalizedRanking operasi) dalam PutEvents permintaan. Amazon Personalize memperoleh tayangan implisit berdasarkan data rekomendasi Anda.

Misalnya, Anda mungkin memiliki aplikasi yang memberikan rekomendasi untuk streaming video. Alur kerja rekomendasi Anda menggunakan tayangan implisit mungkin sebagai berikut:

1. Anda meminta rekomendasi video untuk salah satu pengguna menggunakan operasi Amazon Personalize the section called “GetRecommendations” API.
2. Amazon Personalize menghasilkan rekomendasi untuk pengguna yang menggunakan model Anda (versi solusi) dan mengembalikannya dengan `recommendationId` respons API.
3. Anda menunjukkan rekomendasi video kepada pengguna Anda di aplikasi Anda.

4. Saat pengguna berinteraksi dengan (misalnya, mengklik) video, rekam pilihan tersebut dalam panggilan ke PutEvents API dan sertakan parameter `recommendationId` as a. Untuk contoh kode lihatMerekam peristiwa interaksi item dengan data tayangan.
5. Amazon Personalize menggunakan data tayangan `recommendationId` untuk memperoleh data tayangan dari rekomendasi video sebelumnya, dan kemudian menggunakan data tayangan untuk memandu eksplorasi, di mana rekomendasi future menyertakan video baru dengan data interaksi atau relevansi yang lebih sedikit.

   Untuk informasi selengkapnya tentang merekam peristiwa dengan data tayangan implisit, lihatMerekam peristiwa interaksi item dengan data tayangan.

Contoh data interaksi

Data interaksi berikut mewakili aktivitas pengguna historis dari situs web video streaming. Anda dapat menggunakan data untuk melatih model yang menyediakan rekomendasi film berdasarkan data interaksi pengguna. Perhatikan bahwa beberapa nilai untuk EVENT_VALUE adalah null.

```
USER_ID,ITEM_ID,EVENT_TYPE,EVENT_VALUE,TIMESTAMP
196,242,watch,.50,881250949
186,302,watch,.75,891717742
22,377,click,,878887116
244,51,click,,880606923
166,346,watch,.50,886397596
298,474,watch,.25,884182806
115,265,click,,881171488
253,465,watch,.50,891628467
305,451,watch,.75,886324817
```

Amazon Personalize membutuhkan kolom`USER_ID`,`ITEM_ID`, dan`TIMESTAMP`. `USER_ID`adalah pengenal untuk pengguna aplikasi Anda. `ITEM_ID`adalah pengenal untuk sebuah film. `EVENT_TYPE`dan `EVENT_VALUE` merupakan pengidentifikasi untuk interaksi pengguna. Dalam data sampel, peristiwa adalah `watch` dan `click` peristiwa dan nilainya adalah persentase video yang ditonton pengguna. `TIMESTAMP`Ini mewakili waktu zaman Unix saat pembelian film berlangsung.

Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize. Seperti inilah tampilan file JSON skema untuk data sampel.

```
{
  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
    {
      "name": "USER_ID",
      "type": "string"
    },
    {
      "name": "ITEM_ID",
      "type": "string"
    },
    { "name": "EVENT_TYPE",
      "type": "string"
    },
    {
      "name": "EVENT_VALUE",
      "type": "float"
    },
    {
      "name": "TIMESTAMP",
      "type": "long"
    }
  ],
  "version": "1.0"
}
```

Mempersiapkan metadata item untuk pelatihan

Metadata item mencakup data numerik dan kategoris tentang item yang berinteraksi dengan pengguna Anda. Contoh metadata item termasuk stempel waktu pembuatan, harga, genre, deskripsi, dan ketersediaan. Anda mengimpor metadata tentang item Anda ke dalam kumpulan data Amazon Personalize Items.

Bergantung pada kasus penggunaan domain atau resep khusus Anda, metadata item dapat membantu Amazon Personalize merekomendasikan item yang lebih relevan kepada pengguna, memprediksi item serupa dengan lebih akurat, atau merekomendasikan segmen pengguna yang lebih bermakna. Dan itu dapat membantu Amazon Personalisasi menampilkan item baru dalam rekomendasi. Metadata item diperlukan untuk beberapa kasus penggunaan domain dan opsional untuk semua resep khusus. Untuk informasi selengkapnya, lihat persyaratan data untuk kasus

penggunaan domain atau resep Anda diMencocokkan kasus penggunaan Anda dengan Amazon Personalize resource.

Saat pelatihan, Amazon Personalize tidak menggunakan data item string non-kategoris, seperti judul item atau data penulis. Namun, mengimpor data ini masih dapat meningkatkan rekomendasi. Untuk informasi selengkapnya, lihat Data string non-kategoris.

Jumlah maksimum item yang dipertimbangkan Amazon Personalize selama pelatihan tergantung pada kasus penggunaan atau resep Anda. Hanya item yang dipertimbangkan selama pelatihan yang dapat muncul dalam rekomendasi.

- Untuk User-Personalization-v 2 atau Personalized-Ranking-v 2, jumlah maksimum item yang dipertimbangkan oleh model selama pelatihan adalah 5 juta. Item ini berasal dari kumpulan data interaksi Item dan Item.
- Untuk semua kasus penggunaan domain dan resep khusus selain User-Personalization-v 2 dan Personalized-Ranking-v 2, jumlah maksimum item yang dipertimbangkan oleh model selama pelatihan dan menghasilkan rekomendasi adalah 750.000.

Untuk semua kasus penggunaan domain dan resep khusus, data item massal Anda harus dalam file CSV. Setiap baris dalam file harus mewakili item unik. Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

Bagian berikut memberikan informasi selengkapnya tentang cara menyiapkan metadata item Anda untuk Amazon Personalize. Untuk pedoman format data massal untuk semua jenis data, lihat pedoman format data massal

Topik

- Persyaratan data item
- Data stempel waktu pembuatan
- Metadata kategoris
- Metadata teks tidak terstruktur
- Data numerik
- Data string non-kategoris
- Contoh metadata item

Persyaratan data item

Berikut ini adalah persyaratan metadata item untuk Amazon Personalize.

Jika Anda tidak yakin memiliki cukup data atau jika Anda memiliki pertanyaan tentang kualitasnya, Anda dapat mengimpor data Anda ke kumpulan data Amazon Personalize dan menggunakan Amazon Personalize untuk menganalisisnya. Untuk informasi selengkapnya, lihat Menganalisis kualitas dan kuantitas data di Amazon Personalisasi kumpulan data.

- Untuk semua kasus penggunaan domain dan resep khusus, Anda harus memiliki kolom ITEM_ID yang menyimpan pengenal unik untuk setiap item. Setiap item harus memiliki ID item. Itu harus `string` dengan panjang maksimal 256 karakter.
- Untuk resep kustom, data Anda harus memiliki setidaknya satu string kategoris atau kolom metadata numerik. Kolom metadata item dapat menyertakan empty/null nilai. Kami menyarankan agar kolom-kolom ini minimal 70 persen selesai.
- Untuk kasus penggunaan domain, kolom yang diperlukan bergantung pada domain Anda. Untuk informasi selengkapnya, lihat Persyaratan domain VIDEO_ON_DEMAND atau Persyaratan domain ECOMMERCE.
- Jumlah maksimum kolom metadata adalah 100.

Persyaratan domain VIDEO_ON_DEMAND

Metadata item diperlukan untuk beberapa kasus penggunaan (lihatKasus penggunaan VIDEO_ON_DEMAND). Jika opsional, kami tetap menyarankan untuk mengimpor metadata item untuk mendapatkan rekomendasi yang paling relevan. Jika Anda mengimpor metadata item, data Anda harus menyertakan kolom berikut:

- ITEM_ID
- GENRE (kategoris`string`)
- CREATION_TIMESTAMP (dalam format waktu zaman Unix)

Berikut ini mencantumkan kolom tambahan yang direkomendasikan dan jenis yang diperlukan. `null`Jenis menunjukkan bahwa kolom dapat memiliki nilai yang hilang. Kami menyarankan agar kolom-kolom ini minimal 70 persen selesai. Termasuk kolom ini dapat meningkatkan rekomendasi.

- HARGA (float)

- DURASI (mengapung)
- GENRE_L2 (kategoris,) `string` `null`
- GENRE_L3 (kategoris,) `string` `null`
- RATING_RATA-RATA (,) `float` `null`
- PRODUCT_DESCRIPTION (tekstual,) `string` `null`
- CONTENT_OWNER (kategoris`string`,`null`) — Perusahaan yang memiliki video. Misalnya, nilai mungkin HBO, Paramount, dan NBC.
- CONTENT_CLASSIFICATION (kategoris`string`,`null`) - Peringkat konten. Misalnya, nilai mungkin G, PG, PG-13, R, NC-17, dan tidak dinilai.

Persyaratan domain ECOMMERCE

Metadata item adalah opsional untuk semua kasus penggunaan ECOMMERCE. Jika Anda memiliki data item, kami sarankan untuk mengimpornya untuk mendapatkan rekomendasi yang paling relevan. Jika Anda mengimpor metadata item, data Anda harus memiliki kolom berikut:

- ITEM_ID
- HARGA (`float`)
- CATEGORY_L1 (kategoris`string`) — Untuk informasi tentang memformat data kategoris, lihat. Metadata kategoris

Berikut ini mencantumkan kolom tambahan yang direkomendasikan dan jenis yang diperlukan. `null`Jenis menunjukkan bahwa kolom dapat memiliki nilai yang hilang. Kami menyarankan agar kolom-kolom ini minimal 70 persen selesai. Termasuk kolom ini dapat meningkatkan rekomendasi.

- KATEGORI_L2 (kategoris,) `string` `null`
- CATEGORY_L3 (kategoris,) `string` `null`
- PRODUCT_DESCRIPTION (tekstual,) `string` `null`
- CREATION_TIMESTAMP () `float`
- AGE_GROUP (kategoris`string`,`null`) - Kelompok usia item tersebut untuk. Nilai mungkin bayi baru lahir, bayi, anak-anak, dan orang dewasa.
- DEWASA (kategoris`string`,`null`) — Apakah barang dibatasi hanya untuk orang dewasa, seperti alkohol. Nilai mungkin ya atau tidak.

- GENDER (kategori`string`,`null`) - Jenis kelamin item tersebut. Nilai mungkin laki-laki, perempuan, dan unisex.

Data stempel waktu pembuatan

Data stempel waktu pembuatan harus dalam format waktu epoch Unix dalam hitungan detik. Misalnya, stempel waktu Epoch dalam detik untuk tanggal 31 Juli 2020 adalah 1596238243. Untuk mengonversi tanggal ke stempel waktu epoch Unix, gunakan konverter Epoch - konverter stempel waktu Unix.

Amazon Personalize menggunakan data stempel waktu pembuatan (dalam format waktu epoch Unix, dalam hitungan detik) untuk menghitung usia item dan menyesuaikan rekomendasi yang sesuai.

Jika data stempel waktu pembuatan hilang untuk satu atau beberapa item, Amazon Personalize menyimpulkan informasi ini dari data interaksi, jika ada, dan menggunakan stempel waktu data interaksi tertua item sebagai stempel waktu pembuatan item. Jika item tidak memiliki data interaksi, stempel waktu pembuatannya ditetapkan sebagai stempel waktu interaksi terbaru dalam set pelatihan dan Amazon Personalize menganggapnya sebagai item baru.

Metadata kategoris

Dengan resep tertentu dan semua kasus penggunaan domain, Amazon Personalize menggunakan metadata kategoris, seperti genre atau warna item, saat mengidentifikasi pola dasar yang mengungkapkan item yang paling relevan bagi pengguna Anda. Anda menentukan rentang nilai Anda sendiri berdasarkan kasus penggunaan Anda. Metadata kategoris bisa dalam bahasa apa pun.

Untuk item dengan beberapa kategori, pisahkan setiap nilai dengan bilah vertikal, '|'. Misalnya, untuk bidang GENRES, data Anda untuk item mungkin`Action|Crime|Biopic`. Jika Anda memiliki beberapa level data kategoris dan beberapa item memiliki beberapa kategori untuk setiap level dalam hierarki, gunakan kolom terpisah untuk setiap level dan tambahkan indikator level setelah setiap nama bidang: GENRES, GENRE_L2, GENRE_L3. Ini memungkinkan Anda untuk memfilter rekomendasi berdasarkan sub-kategori, bahkan jika item termasuk dalam beberapa kategori multi-level (untuk informasi tentang membuat dan menggunakan filter lihatMemfilter rekomendasi dan segmen pengguna). Misalnya, video mungkin memiliki data berikut untuk setiap tingkat kategori:

- GENRE: Aksi|Petualangan
- GENRE_L2: Kejahatan|Barat
- GENRE_L3: Biopik

Dalam contoh ini, video berada dalam hierarki aksi> kriminal> biopik dan hierarki biopik petualangan> barat>. Kami merekomendasikan hanya menggunakan hingga L3 tetapi Anda dapat menggunakan lebih banyak level jika perlu.

Nilai kategoris dapat memiliki maksimal 1000 karakter. Jika Anda memiliki item dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal. Kami merekomendasikan kolom kategoris memiliki paling banyak 1000 nilai yang mungkin. Mengimpor data kategoris dengan lebih banyak nilai dapat berdampak negatif pada rekomendasi. Berikut ini dapat membantu Anda mengurangi jumlah nilai yang mungkin untuk kolom kategoris:

- Pastikan nilai mengikuti konvensi penamaan yang konsisten dan periksa kesalahan ketik. Misalnya, gunakan “Sepatu Pria” daripada memiliki campuran “Sepatu Pria”, “Sepatu Pria”, dan “Sepatu Pria”.
- Konsolidasikan kategori serupa yang menggunakan istilah yang sedikit berbeda mengacu pada kategori dasar yang sama, seperti “Sepatu” dan “Sepatu Kets”.
- Jika data Anda memiliki struktur hierarkis, di mana kategori yang lebih luas (seperti “Alas Kaki”) berisi subkategori yang lebih spesifik (seperti “Sepatu Pria”, “Sepatu Wanita”, “Sepatu Anak”), gunakan kolom terpisah untuk setiap level dan tambahkan indikator level setelah setiap nama bidang. Misalnya, CATEGORY_1, CATEGORY_2, dan CATEGORY_3. Ini dapat mengurangi kategori yang ambigu atau tumpang tindih.

Dengan semua resep dan domain, Anda dapat mengimpor data kategoris dan menggunakannya untuk memfilter rekomendasi berdasarkan atribut item. Untuk informasi tentang rekomendasi penyaringan, lihatMemfilter rekomendasi dan segmen pengguna.

Metadata teks tidak terstruktur

Dengan resep dan domain tertentu, Amazon Personalize dapat mengekstrak informasi yang bermakna dari metadata teks yang tidak terstruktur, seperti deskripsi produk, ulasan produk, atau sinopsis film. Amazon Personalize menggunakan teks tidak terstruktur untuk mengidentifikasi item yang relevan bagi pengguna Anda, terutama ketika item baru atau memiliki lebih sedikit data interaksi. Anda dapat menambahkan paling banyak 1 bidang tekstual. Sertakan data teks tidak terstruktur dalam kumpulan data Item Anda untuk meningkatkan rasio klik-tayang dan rasio percakapan untuk item baru di katalog Anda.

Saat Anda menyiapkan metadata teks tidak terstruktur, bungkus teks dengan tanda kutip ganda dan hapus karakter baris baru. Gunakan \ karakter untuk menghindari tanda kutip ganda atau\ karakter

dalam data Anda. Amazon Personalize memotong bidang teks pada batas karakter. Pastikan bahwa informasi yang paling relevan dalam teks ada di awal bidang.

Nilai teks yang tidak terstruktur dapat memiliki paling banyak 20.000 karakter dalam semua bahasa kecuali bahasa Mandarin dan Jepang. Untuk bahasa Mandarin dan Jepang, Anda dapat memiliki paling banyak 7.000 karakter. Amazon Personalize memotong nilai yang melebihi batas karakter hingga batas karakter.

Anda dapat mengirimkan item teks tidak terstruktur dalam beberapa bahasa, tetapi teks setiap item harus hanya dalam satu bahasa. Teks dapat dalam bahasa berikut:

- Mandarin (Sederhana)
- Mandarin (Tradisional)
- Bahasa Inggris
- Bahasa Prancis
- Bahasa Jerman
- Bahasa Jepang
- Bahasa Portugis
- Bahasa Spanyol

Data numerik

Amazon Personalize dapat menggunakan metadata item numerik, seperti harga atau durasi video, untuk menghasilkan rekomendasi yang lebih relevan bagi pengguna. Data numerik ini dapat direpresentasikan sebagai bilangan bulat atau nilai desimal.

Jika Anda menggunakan Personalisasi Pengguna atau resep Peringkat yang Dipersonalisasi khusus, Anda dapat mengoptimalkan solusi Amazon Personalisasi untuk tujuan terkait metadata Item selain relevansi maksimum, seperti memaksimalkan pendapatan. Saat mengonfigurasi solusi, Anda memilih kolom metadata numerik di kumpulan data Item yang terkait dengan tujuan Anda. Misalnya, Anda dapat memilih kolom VIDEO_LENGTH untuk memaksimalkan menit streaming atau kolom PRICE untuk memaksimalkan pendapatan.

Untuk informasi selengkapnya, lihat Mengoptimalkan solusi untuk tujuan tambahan.

Data string non-kategoris

Kecuali untuk item IDs, Amazon Personalize tidak menggunakan data string non-tekstual non-kategoris saat pelatihan, seperti judul item atau data penulis. Namun, Amazon Personalize dapat menggunakannya dengan fitur-fitur berikut. Nilai non-kategoris dapat memiliki maksimum 1000 karakter.

- Amazon Personalize dapat menyertakan metadata item dalam rekomendasi, termasuk nilai string non-kategoris. Anda dapat menggunakan metadata untuk memperkaya rekomendasi di antarmuka pengguna, seperti menambahkan nama sutradara ke korsel rekomendasi film. Untuk informasi selengkapnya, lihat Metadata item dalam rekomendasi.
- Jika Anda menggunakanBarang Serupa, Anda dapat menghasilkan rekomendasi batch dengan tema. Saat Anda membuat rekomendasi batch dengan tema, Anda harus menentukan kolom nama item dalam pekerjaan inferensi batch. Untuk informasi selengkapnya, lihat Rekomendasi Batch dengan tema dari Content Generator.
- Anda dapat membuat filter untuk menyertakan atau menghapus item dari rekomendasi berdasarkan data string non-kategoris. Untuk informasi lebih lanjut tentang filter, lihat Memfilter rekomendasi dan segmen pengguna.

Contoh metadata item

Beberapa baris pertama metadata film dalam file CSV mungkin terlihat seperti berikut.

```
ITEM_ID,GENRES,CREATION_TIMESTAMP,DESCRIPTION
1,Adventure|Animation|Children|Comedy|Fantasy,1570003267,"This is an animated movie
 that features action, comedy, and fantasy. Audience is children. This movie was
 released in 2004."
2,Adventure|Children|Fantasy,1571730101,"This is an adventure movie with elements of
 fantasy. Audience is children. This movie was release in 2010."
3,Comedy|Romance,1560515629,"This is a romantic comedy. The movie was released in 1999.
 Audience is young women."
4,Comedy|Drama|Romance,1581670067,"This movie includes elements of both comedy and
 drama as well as romance. This movie was released in 2020."
...
...
```

`ITEM_ID`Kolom diperlukan dan menyimpan pengidentifikasi unik untuk setiap item individu. `GENRE`Kolom menyimpan metadata kategoris untuk setiap film dan `DESCRIPTION` kolom adalah

metadata tekstual tidak terstruktur. `CREATION_TIMESTAMP`Kolom menyimpan setiap waktu pembuatan item dalam format waktu epoch Unix dalam hitungan detik.

Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize. Seperti inilah tampilan file JSON skema untuk data sampel di atas.

```
{
  "type": "record",
  "name": "Items",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
    {
      "name": "ITEM_ID",
      "type": "string"
    },
    {
      "name": "GENRES",
      "type": [
        "null",
        "string"
      ],
      "categorical": true
    },
    {
      "name": "CREATION_TIMESTAMP",
      "type": "long"
    },
    {
      "name": "DESCRIPTION",
      "type": [
        "null",
        "string"
      ],
      "textual": true
    }
  ],
  "version": "1.0"
}
```

Mempersiapkan metadata pengguna untuk pelatihan

Data pengguna yang dapat Anda impor ke Amazon Personalize mencakup data numerik, seperti usia pengguna, dan metadata kategoris, seperti jenis kelamin atau keanggotaan loyalitas. Anda mengimpor metadata tentang pengguna Anda ke dalam kumpulan data Amazon Personalize Users.

Bergantung pada kasus penggunaan domain atau resep khusus Anda, metadata pengguna dapat membantu Amazon Personalize merekomendasikan item yang lebih relevan kepada pengguna atau merekomendasikan segmen pengguna yang lebih bermakna. Dan setelah pelatihan, ini dapat membantu model Anda merekomendasikan item untuk pengguna tanpa data interaksi apa pun. Untuk informasi selengkapnya tentang kasus penggunaan atau resep yang menggunakan metadata pengguna, lihat persyaratan data untuk kasus atau resep penggunaan domain Anda. Mencocokkan kasus penggunaan Anda dengan Amazon Personalize resource

Saat pelatihan, Amazon Personalize tidak menggunakan data pengguna string non-kategoris, seperti nama pengguna, kata kunci tentang pengguna, atau tag. Namun, mengimpor data ini masih dapat meningkatkan rekomendasi. Untuk informasi selengkapnya, lihat Data string non-kategoris.

Untuk semua kasus penggunaan domain dan resep khusus, data pengguna massal Anda harus dalam file CSV. Setiap baris dalam file harus mewakili pengguna yang unik. Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

Bagian berikut memberikan informasi selengkapnya tentang cara menyiapkan data pengguna Anda untuk Amazon Personalize. Untuk pedoman format data massal untuk semua jenis data, lihat pedoman format data massal

Topik

- Persyaratan data pengguna
- Metadata kategoris
- Data string non-kategoris
- Contoh metadata pengguna

Persyaratan data pengguna

Berikut ini adalah persyaratan data pengguna untuk Amazon Personalize. Anda bebas menambahkan kolom kustom tambahan tergantung pada kasus penggunaan dan data Anda.

- Data Anda harus memiliki kolom USER_ID yang menyimpan pengenal unik untuk setiap pengguna. Setiap pengguna harus memiliki ID pengguna. Itu harus `string` dengan panjang maksimal 256 karakter.
- Data Anda harus memiliki setidaknya satu string kategoris atau kolom metadata numerik. Kolom metadata pengguna dapat menyertakan empty/null nilai untuk beberapa pengguna. Kami menyarankan agar kolom-kolom ini minimal 70 persen selesai.
- Jumlah maksimum kolom metadata adalah 25.

Jika Anda tidak yakin memiliki cukup data atau jika Anda memiliki pertanyaan tentang kualitasnya, Anda dapat mengimpor data Anda ke kumpulan data Amazon Personalize dan menggunakan Amazon Personalize untuk menganalisisnya. Untuk informasi selengkapnya, lihat Menganalisis kualitas dan kuantitas data di Amazon Personalisasi kumpulan data.

Metadata kategoris

Dengan beberapa resep dan semua kasus penggunaan domain, Amazon Personalize menggunakan metadata kategoris, seperti jenis kelamin, minat, atau status keanggotaan pengguna, saat mengidentifikasi pola dasar yang mengungkapkan item paling relevan bagi pengguna Anda. Anda menentukan rentang nilai Anda sendiri berdasarkan kasus penggunaan Anda. Metadata kategoris bisa dalam bahasa apa pun.

Untuk pengguna dengan beberapa kategori, pisahkan setiap nilai dengan bilah vertikal, '|'. Misalnya, untuk bidang KEPENTINGAN, data Anda untuk pengguna mungkin`Movies|TV Shows|Music`.

Dengan semua resep dan domain, Anda dapat mengimpor metadata kategoris dan menggunakannya untuk memfilter rekomendasi berdasarkan atribut pengguna. Untuk informasi tentang rekomendasi pemfilteran lihatMemfilter rekomendasi dan segmen pengguna.

Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki pengguna dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Data string non-kategoris

Kecuali untuk pengguna IDs, Amazon Personalize tidak menggunakan data string non-kategoris saat pelatihan, seperti nama pengguna, kata kunci tentang pengguna, atau tag. Namun, Amazon Personalize dapat menggunakannya saat memfilter rekomendasi. Anda dapat membuat filter untuk menyertakan atau menghapus item dari rekomendasi berdasarkan data string non-kategoris tentang pengguna yang Anda dapatkan rekomendasi untuk (the). CurrentUser Untuk informasi lebih lanjut

tentang filter, lihat Memfilter rekomendasi dan segmen pengguna. Nilai non-kategoris dapat memiliki maksimum 1000 karakter.

Contoh metadata pengguna

Beberapa baris pertama metadata pengguna dalam file CSV mungkin terlihat seperti berikut ini.

```
USER_ID,AGE,GENDER,INTEREST
5,34,Male,hiking
6,56,Female,music
8,65,Male,movies|TV shows|music
...
...
```

`USER_ID`Kolom diperlukan dan menyimpan pengidentifikasi unik untuk setiap pengguna individu. `AGE`Kolom adalah metadata numerik. `INTEREST`Kolom `GENDER` dan menyimpan metadata kategoris untuk setiap pengguna.

Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize. Seperti inilah tampilan file JSON skema untuk data sampel di atas.

```
{
  "type": "record",
  "name": "Users",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "AGE",
          "type": "int"
      },
      {
          "name": "GENDER",
          "type": "string",
          "categorical": true
      },
      {
```

```
            "name": "INTEREST",
            "type": "string",
            "categorical": true
        }
    ],
    "version": "1.0"
}
```

Mempersiapkan metadata tindakan untuk pelatihan

Tindakan adalah aktivitas penghasil keterlibatan atau pendapatan yang mungkin ingin Anda rekomendasikan kepada pengguna Anda. Tindakan mungkin termasuk menginstal aplikasi seluler Anda, melengkapi profil keanggotaan, bergabung dengan program loyalitas Anda, atau mendaftar untuk email promosi. Anda mengimpor data tentang tindakan Anda ke dalam kumpulan data Amazon Personalize Actions. Contoh data untuk tindakan termasuk ID unik untuk tindakan, nilai estimasi tindakan, atau stempel waktu kedaluwarsa tindakan.

Jika Anda menggunakan Next-Best-Action, Anda harus mengimpor metadata tindakan. Dengan resep ini, Amazon Personalize memprediksi tindakan terbaik berikutnya dari tindakan yang Anda impor ke kumpulan data Tindakan. Tidak ada resep atau kasus penggunaan lain yang menggunakan metadata tindakan. Anda tidak dapat membuat kumpulan data Tindakan dalam grup kumpulan data domain.

Saat pelatihan, Amazon Personalize tidak menggunakan data tindakan string non-kategoris, seperti judul tindakan atau tag. Namun, mengimpor data ini masih dapat meningkatkan rekomendasi. Untuk informasi selengkapnya, lihat Data string non-kategoris.

Data tindakan massal Anda harus dalam file CSV. Setiap baris dalam file harus mewakili tindakan unik. Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

Bagian berikut memberikan informasi selengkapnya tentang cara menyiapkan metadata tindakan Anda untuk Amazon Personalize. Untuk pedoman format data massal untuk semua jenis data, lihat pedoman format data massal

Topik

- Persyaratan data tindakan
- Data stempel waktu kedaluwarsa tindakan

- Ulangi data frekuensi
- Data nilai
- Data stempel waktu pembuatan
- Metadata kategoris
- Data string non-kategoris
- Contoh metadata tindakan

Persyaratan data tindakan

Berikut ini adalah persyaratan data tindakan untuk Amazon Personalize.

- Anda harus memiliki kolom ACTION_ID yang menyimpan pengenal unik untuk setiap tindakan. Setiap tindakan harus memiliki ID item. Itu harus `string` dengan panjang maksimal 256 karakter.
- Data Anda harus memiliki setidaknya satu string kategoris atau kolom metadata numerik. Kolom metadata tindakan dapat menyertakan empty/null nilai. Kami menyarankan agar kolom-kolom ini minimal 70 persen selesai.
- Selama pelatihan model, Amazon Personalize mempertimbangkan maksimum 1000 tindakan. Jika Anda mengimpor lebih dari 1000 tindakan, Amazon Personalize memutuskan tindakan mana yang akan disertakan dalam pelatihan, dengan prioritas diberikan pada tindakan baru (tindakan yang baru-baru ini Anda tambahkan tanpa interaksi) dan tindakan yang ada dengan data interaksi terbaru.
- Jumlah kolom maksimum adalah 10.

Data stempel waktu kedaluwarsa tindakan

Stempel waktu kedaluwarsa tindakan menentukan tanggal di mana suatu tindakan tidak lagi valid. Anda menyediakan data stempel waktu kedaluwarsa tindakan dalam format waktu epoch Unix, dalam hitungan detik. Jika suatu tindakan telah kedaluwarsa, Amazon Personalize tidak akan memasukkannya ke dalam rekomendasi.

Tentukan stempel waktu kedaluwarsa tindakan untuk tindakan Anda jika Anda ingin membatasi penampilannya dalam rekomendasi ke kerangka waktu tertentu. Misalnya, Anda mungkin memiliki aplikasi yang menjalankan drive keanggotaan selama bulan tertentu. Anda dapat menetapkan stempel waktu kedaluwarsa untuk tindakan pendaftaran untuk akhir bulan itu. Amazon Personalize secara otomatis berhenti merekomendasikan tindakan ini ketika tanggal ini tercapai.

Jika Anda menyetel stempel waktu kedaluwarsa ke waktu di masa lalu untuk tindakan baru, atau jika Anda memperbarui stempel waktu tindakan ke waktu sebelumnya, diperlukan waktu hingga 2 jam untuk menghapus tindakan dari rekomendasi.

Ulangi data frekuensi

Data frekuensi berulang menentukan berapa hari Amazon Personalize harus menunggu untuk merekomendasikan tindakan tertentu setelah pengguna berinteraksi dengannya, berdasarkan riwayat pengguna dalam kumpulan data interaksi Tindakan Anda. Anda menentukan frekuensi pengulangan tindakan dalam beberapa hari, dengan maksimum 30.

Misalnya, Anda mungkin memiliki aplikasi e-commerce di mana setiap pengguna membuat akun dan profil. Jika Anda memiliki `complete profile` tindakan dan Anda ingin menunggu seminggu setelah pengguna berinteraksi dengannya sebelum merekomendasikannya lagi, Anda akan menentukan 7 hari sebagai tindakan. `REPEAT_FREQENCY` Setelah 7 hari, Amazon Personalize mulai mempertimbangkan tindakan untuk rekomendasi.

Jika Anda tidak memberikan frekuensi pengulangan untuk suatu tindakan, Amazon Personalize tidak akan menetapkan batasan berapa kali muncul dalam rekomendasi.

Data nilai

Data nilai adalah nilai bisnis atau pentingnya setiap tindakan. Tindakan `value` dapat berupa 1 — 10, di mana 10 adalah tindakan paling berharga dalam kumpulan data Anda.

Misalnya, Anda mungkin memiliki dua tindakan, satu untuk mendaftar di langganan dasar Anda dan satu untuk mendaftar di layanan premium Anda. Untuk layanan dasar, Anda dapat menentukan nilai `5` dan untuk premi, nilai`10`.

Amazon Personalize menggunakan data nilai sebagai satu masukan saat menentukan tindakan terbaik untuk direkomendasikan kepada pengguna Anda. Misalnya, jika pengguna memiliki kemungkinan yang sama untuk mengambil satu tindakan atau lainnya, Amazon Personalize memberi peringkat tindakan dengan nilai tertinggi yang lebih tinggi dalam rekomendasi.

Data stempel waktu pembuatan

Amazon Personalize menggunakan data stempel waktu pembuatan (dalam format waktu epoch Unix, dalam hitungan detik) untuk menghitung usia tindakan dan menyesuaikan rekomendasi yang sesuai.

Jika Anda tidak memiliki data stempel waktu pembuatan, Amazon Personalize menyimpulkan informasi ini dari data interaksi tindakan apa pun. Ini menggunakan stempel waktu data interaksi tertua tindakan sebagai stempel waktu pembuatan tindakan. Jika suatu tindakan tidak memiliki data interaksi, stempel waktu pembuatannya ditetapkan sebagai stempel waktu interaksi terbaru dalam set pelatihan, dan Amazon Personalize menganggapnya sebagai tindakan baru.

Metadata kategoris

Amazon Personalize menggunakan metadata kategoris tentang tindakan, seperti musiman atau eksklusivitas tindakan, saat mengidentifikasi pola dasar yang mengungkapkan tindakan terbaik bagi pengguna Anda. Anda menentukan rentang nilai Anda sendiri berdasarkan kasus penggunaan Anda. Metadata kategoris bisa dalam bahasa apa pun.

Anda dapat mengimpor data kategoris dan menggunakannya untuk memfilter rekomendasi berdasarkan atribut tindakan. Untuk informasi tentang rekomendasi penyaringan, lihatMemfilter rekomendasi dan segmen pengguna.

Nilai kategoris dapat memiliki maksimal 1000 karakter. Jika Anda memiliki tindakan dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Data string non-kategoris

Kecuali untuk tindakan IDs, Amazon Personalize tidak menggunakan data string non-kategoris saat pelatihan, seperti nama tindakan, kata kunci tentang tindakan, atau tag. Namun, Amazon Personalize dapat menggunakannya saat memfilter rekomendasi. Anda dapat membuat filter untuk menyertakan atau menghapus tindakan dari rekomendasi berdasarkan data string non-kategoris. Untuk informasi lebih lanjut tentang filter, lihat Memfilter rekomendasi dan segmen pengguna. Nilai non-kategoris dapat memiliki maksimum 1000 karakter.

Contoh metadata tindakan

Beberapa baris pertama metadata tindakan dalam file CSV mungkin terlihat seperti berikut.

```
ACTION_ID,VALUE,MEMBERSHIP_LEVEL,CREATION_TIMESTAMP,REPEAT_FREQUENCY
1,10,Deluxe|Premium,1510003267,7
2,5,Basic,1580003267,7
3,5,Preview,1590003267,3
4,10,Deluxe|Platinum,1560003267,4
...
```

```
...
```

ACTION_IDKolom diperlukan. MEMBERSHIP_LEVELKolom adalah bidang string kategoris. REPEAT_FREQUENCYBidang VALUECREATION_TIMESTAMP,, dan adalah kata kunci yang dicadangkan dengan jenis yang diperlukan.

Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize. Seperti inilah tampilan file JSON skema untuk data sampel di atas.

```
{
  "type": "record",
  "name": "Actions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
    {
      "name": "ACTION_ID",
      "type": "string"
    },
    {
      "name": "VALUE",
      "type": [
        "null",
        "long"
      ]
    },

    {
      "name": "MEMBERSHIP_LEVEL",
      "type": [
        "null",
        "string"
      ],
      "categorical": true
    },

    {
      "name": "CREATION_TIMESTAMP",
      "type": "long"
    },
    {
      "name": "REPEAT_FREQUENCY",
```

```
      "type": [
        "long",
        "null"
      ]
    }
  ],
  "version": "1.0"
}
```

Mempersiapkan data interaksi tindakan untuk pelatihan

Jika Anda menggunakan resep kustom Next-Best-Action, Amazon Personalize menggunakan data interaksi tindakan untuk mengidentifikasi minat pengguna dan memprediksi tindakan yang kemungkinan besar akan mereka lakukan. Interaksi aksi adalah interaksi yang melibatkan pengguna dan tindakan dalam kumpulan data Tindakan Anda. Misalnya, jika Anda memiliki tindakan pendaftaran di kumpulan data Tindakan, dan pengguna melakukan tindakan ini, Anda akan merekam ID pengguna, ID tindakan, stempel waktu, dan untuk jenis peristiwa, catatan. `TAKEN`

Anda mengimpor interaksi tindakan ke dalam kumpulan data interaksi Amazon Personalize Action. Anda dapat mengimpor peristiwa interaksi tindakan secara massal dengan tugas impor kumpulan data, atau Anda dapat mengalirkannya secara real time dengan operasi PutActionInteractions API. Anda tidak dapat membuat sumber daya tindakan terbaik berikutnya, termasuk kumpulan data Tindakan dan Interaksi Tindakan, dalam grup kumpulan data domain.

Data interaksi tindakan massal Anda harus dalam file CSV. Setiap baris dalam file harus mewakili interaksi unik antara pengguna dan tindakan. Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

Bagian berikut memberikan informasi selengkapnya tentang cara menyiapkan data interaksi tindakan Anda untuk Amazon Personalisasi. Untuk pedoman format data massal untuk semua jenis data, lihat pedoman format data massal.

Topik

- Persyaratan data interaksi tindakan
- Data tipe acara
- Contoh data interaksi aksi

Persyaratan data interaksi tindakan

Tidak ada persyaratan minimum untuk data interaksi tindakan. Kami menyarankan Anda mengimpornya untuk rekomendasi tindakan kualitas. Jika Anda tidak memiliki data interaksi tindakan, Anda dapat membuat kumpulan data interaksi Tindakan kosong dan merekam interaksi pelanggan Anda dengan tindakan menggunakan operasi PutActionInteractions API.

Data interaksi tindakan Anda harus memiliki setidaknya kolom berikut. Anda bebas menambahkan kolom kustom tambahan tergantung pada kasus penggunaan dan data Anda.

- USER_ID — Pengidentifikasi unik dari pengguna yang berinteraksi dengan item. Setiap acara harus memiliki USER_ID. Itu harus `string` dengan panjang maksimal 256 karakter.
- ACTION_ID — Pengidentifikasi unik dari item yang berinteraksi dengan pengguna. Setiap acara harus memiliki ID item. Itu harus `string` dengan panjang maksimal 256 karakter.
- TIMESTAMP — Waktu kejadian terjadi (dalam format waktu epoch Unix dalam hitungan detik). Setiap interaksi tindakan harus memiliki TIMESTAMP. Untuk informasi selengkapnya, lihat Data stempel waktu.
- EVENT_TYPE — Apakah tindakan diambil, tidak diambil, atau dilihat. Setiap interaksi tindakan harus memiliki jenis acara. Untuk informasi selengkapnya, lihat Data tipe acara.

Hingga Anda mengimpor data interaksi tindakan, Amazon Personalize merekomendasikan tindakan tanpa personalisasi, dan skor kecenderungan adalah 0,0. Suatu tindakan akan memiliki skor setelah tindakan memiliki yang berikut:

- Setidaknya 50 interaksi aksi dengan tipe acara TAKEN.
- Setidaknya 50 interaksi tindakan dengan jenis acara NOT_TAKEN atau VIEWED.

Interaksi aksi ini harus ada pada pelatihan versi solusi terbaru, dan harus terjadi dalam rentang waktu 6 minggu dari stempel waktu interaksi terbaru dalam kumpulan data interaksi Tindakan.

Data tipe acara

Amazon Personalize dapat menggunakan pola dalam data tipe peristiwa untuk mengidentifikasi tindakan yang kemungkinan besar akan dilakukan pengguna Anda. Misalnya, jika pelanggan sering mengabaikan tindakan berlangganan email (ditunjukkan dengan jenis peristiwa NOT_TAKEN), Amazon Personalize mungkin menyesuaikan rekomendasi untuk menampilkan lebih sedikit jenis tindakan ini.

Anda hanya dapat menggunakan jenis peristiwa berikut untuk peristiwa interaksi tindakan. Amazon Personalize menggunakan peristiwa ini untuk mempelajari tentang pengguna Anda dan menghitung tindakan apa yang akan direkomendasikan selanjutnya.

- Diambil - Rekam peristiwa yang diambil saat pengguna mengambil tindakan yang disarankan.
- Tidak Diambil - Rekam peristiwa yang Tidak Diambil saat pengguna Anda membuat pilihan yang disengaja untuk tidak mengambil tindakan setelah melihatnya. Misalnya, jika mereka memilih Tidak ketika Anda menunjukkan tindakan kepada mereka. Acara yang Tidak Diambil dapat menunjukkan bahwa pelanggan tidak tertarik dengan tindakan tersebut.
- Dilihat — Rekam peristiwa yang Dilihat saat Anda menunjukkan tindakan kepada pengguna sebelum mereka membuat pilihan untuk mengambil atau tidak mengambil tindakan. Amazon Personalize menggunakan Lihat peristiwa untuk mempelajari minat pengguna Anda. Misalnya, jika pengguna melihat tindakan tetapi tidak menerimanya, pengguna ini mungkin tidak tertarik dengan tindakan ini di masa mendatang.

Contoh data interaksi aksi

Beberapa baris pertama dari file CSV dengan data interaksi tindakan dan semua kolom yang diperlukan mungkin terlihat seperti berikut.

```
USER_ID,ACTION_ID,EVENT_TYPE,TIMESTAMP
35,73,Viewed,1586731606
54,35,Not taken,1586731609
9,33,Viewed,1586735158
23,10,Taken,1586735697
27,11,Taken,1586735763
...
...
```

Setelah Anda selesai mempersiapkan data Anda, Anda siap untuk membuat file JSON skema. File ini memberi tahu Amazon Personalize tentang struktur data Anda. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize. Seperti inilah tampilan file JSON skema untuk data sampel di atas.

```
{

  "type": "record",
  "name": "ActionInteractions",
```

```
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "ACTION_ID",
          "type": "string"
      },
      {
          "name": "EVENT_TYPE",
          "type": "string"
      },
      {
          "name": "TIMESTAMP",
          "type": "long"
      }
  ],
  "version": "1.0"
}
```

Membuat file JSON skema untuk skema Amazon Personalize

Setelah Anda menyiapkan data Anda, Anda siap untuk membuat file JSON skema untuk setiap jenis data yang Anda impor. File-file ini menguraikan struktur dan konten data Anda, termasuk nama kolom dan tipe datanya.

Anda menggunakan file JSON skema saat membuat skema Amazon Personalize di. Membuat skema dan dataset Di Amazon Personalize, skema adalah sumber daya yang memungkinkan Amazon Personalize mengurai data saat Anda mengimpornya ke kumpulan data Anda. Anda membuat skema untuk setiap kumpulan data yang Anda gunakan.

Untuk sumber daya khusus, setiap kumpulan data memiliki persyaratan skema khusus. Untuk grup kumpulan data Domain, domain yang Anda pilih menentukan kumpulan data dan persyaratan skema Anda. Setiap domain memiliki skema default untuk setiap jenis dataset. Saat membuat kumpulan data, Anda dapat menggunakan skema domain yang ada atau membuat yang baru dengan memodifikasi skema default yang ada. Gunakan skema default sebagai panduan untuk data apa yang akan diimpor untuk domain Anda.

Bagian berikut menyediakan persyaratan kustom dan domain untuk membuat file JSON skema untuk setiap jenis dataset.

Topik

- Persyaratan pemformatan skema
- Kumpulan data dan skema VIDEO_ON_DEMAND
- Kumpulan data dan skema ECOMMERCE
- Kumpulan data dan skema khusus

Persyaratan pemformatan skema

Saat membuat skema untuk kumpulan data dalam grup kumpulan data Domain atau grup kumpulan data kustom, Anda harus mengikuti panduan berikut:

- Anda harus menentukan skema dalam format Avro. Untuk informasi tentang tipe data Avro yang kami dukung, lihatJenis data skema.
- Skema memiliki kunci nama yang nilainya harus cocok dengan tipe dataset.

- Bidang skema dapat muncul dalam urutan apa pun, tetapi harus sesuai dengan urutan header kolom yang sesuai dalam file CSV Anda.
- Skema harus berupa file JSON datar tanpa struktur bersarang. Misalnya, bidang tidak dapat menjadi induk dari beberapa sub-bidang.
- Skema Amazon Personalize tidak mendukung tipe kompleks seperti array dan peta.
- Bidang skema harus memiliki nama alfanumerik yang unik. Misalnya, Anda tidak dapat menambahkan `GENRES_FIELD_1` bidang dan `GENRESFIELD1` bidang.
- Anda harus mendefinisikan bidang wajib sebagai tipe data yang diperlukan. Bidang string kategoris yang dicadangkan harus memiliki `categorical` atribut yang disetel ke`true`, sedangkan bidang string cadangan tidak dapat dikategorikan. Kata kunci tidak bisa ada dalam data Anda.
- Jika Anda menambahkan bidang metadata Anda sendiri dari jenis `string` dan Anda ingin Amazon Personalize menggunakannya saat pelatihan, itu harus menyertakan `categorical` atribut atau atribut (hanya skema Item `textual` yang mendukung bidang dengan atribut tekstual).
- Amazon Personalize tidak menggunakan data `boolean` tipe saat melatih atau memfilter rekomendasi. Agar Amazon Personalisasi menggunakan data boolean saat melatih atau memfilter, gunakan bidang tipe String dan gunakan nilai `"True"` dan `"False"` data Anda. Atau Anda dapat menggunakan tipe int atau panjang dan nilai `0` dan`1`.
- Bidang tekstual harus dari tipe `string` dan harus memiliki `textual` atribut yang disetel ke`true`. Untuk informasi selengkapnya tentang data teks tidak terstruktur, lihatMetadata teks tidak terstruktur.

Kumpulan data grup kumpulan data domain memiliki persyaratan tambahan berdasarkan jenis domain dan kumpulan data. Kumpulan data grup kumpulan data khusus memiliki persyaratan tambahan tergantung pada jenisnya.

Jenis data skema

Skema Amazon Personalize mendukung jenis Avro berikut untuk bidang:

- float
- double
- int
- long
- string
- boolean

- null

Beberapa bidang wajib dan cadangan mendukung data nol. Menambahkan `null` tipe ke bidang memungkinkan Anda menggunakan data yang tidak sempurna (misalnya, metadata dengan nilai kosong) untuk menghasilkan rekomendasi. Untuk informasi tentang bidang mana yang mendukung data nol, lihat topik persyaratan skema untuk domain Anda:Kumpulan data dan skema VIDEO_ON_DEMAND,Kumpulan data dan skema ECOMMERCE, atau. Kumpulan data dan skema khusus Contoh berikut menunjukkan cara menambahkan tipe null untuk bidang GENDER.

```
{
  "name": "GENDER",
  "type": [
    "null",
    "string"
  ],
  "categorical": true
}
```

Kumpulan data dan skema VIDEO_ON_DEMAND

Saat Anda membuat grup kumpulan data Domain untuk domain VIDEO_ON_DEMAND, setiap jenis kumpulan data memiliki skema default dengan sekumpulan bidang wajib dan direkomendasikan khusus VIDEO_ON_DEMAND. Anda dapat menggunakan skema default atau membuat yang baru berdasarkan skema default. Data yang Anda impor harus sesuai dengan skema Anda dalam format dan jenis. Gunakan skema domain default yang tercantum di bagian di bawah ini sebagai panduan untuk menentukan data apa yang akan diimpor untuk membuat pemberi rekomendasi berbasis Video_ON_DEMAND Anda.

Anda bebas menambahkan bidang tambahan. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda.

Untuk informasi tentang persyaratan skema Amazon Personalisasi umum, seperti persyaratan pemformatan dan tipe data bidang yang tersedia, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema, terlepas dari domain.

Topik berikut memberikan informasi tentang bidang wajib dan direkomendasikan setiap kumpulan data untuk domain VIDEO_ON_DEMAND. Setiap bagian kumpulan data menyertakan skema VIDEO_ON_DEMAND default dalam format JSON.

Topik

- Dataset domain VIDEO_ON_DEMAND dan persyaratan skema
- Persyaratan kumpulan data interaksi item (domain VIDEO_ON_DEMAND)
- Persyaratan kumpulan data pengguna (domain VIDEO_ON_DEMAND)
- Persyaratan kumpulan data item (domain VIDEO_ON_DEMAND)

Dataset domain VIDEO_ON_DEMAND dan persyaratan skema

Setiap jenis dataset memiliki bidang wajib berikut dan kata kunci yang dicadangkan. Kata kunci yang dicadangkan adalah opsional, bidang non-metadata. Bidang ini dianggap dicadangkan karena Anda harus menentukan bidang sebagai tipe data yang diperlukan saat Anda menggunakannya. Bidang string kategoris yang dicadangkan harus `categorical` disetel `ketrue`, sementara bidang string yang dicadangkan tidak dapat dikategorikan. Kata kunci tidak bisa ada dalam data Anda.

Jenis dataset	Bidang yang diperlukan	Kata kunci terpesan
Interaksi item (skema default)	USER_ID () `string` ITEM_ID () `string` STEMPEL WAKTU () `long` EVENT_TYPE (`string`dan tergantung pada kasus penggunaan, `Watch` dan `Click` jenis acara)	EVENT_VALUE (,) `float null` KESAN (`string`,`null`) REKOMENDASI_ID (,) `string null` EVENT_ATTRIBUTION_ SOURCE (,) `string null`
Pengguna (skema default)	USER_ID () `string` 1 bidang metadata (kategoris `string` atau numerik)	SUBSCRIPTION_MODEL (kategoris,) `string null`
Item (skema default)	ITEM_ID () `string` CREATION_TIMESTAMP () `long` GENRE (kategoris`string`)	HARGA (`float`,`null`) DURASI (`float`,`null`) GENRE_L2 (kategoris,) `string null`

Jenis dataset	Bidang yang diperlukan	Kata kunci terpesan
		GENRE_L3 (kategoris,) `string null` RATING_RATA-RATA (,) `float null` PRODUCT_DESCRIPTION (tekstual,) `string null` CONTENT_OWNER (kategoris,) `string null` CONTENT_CLASSIFICATION (kategoris,) `string null`

Persyaratan kumpulan data interaksi item (domain VIDEO_ON_DEMAND)

Kumpulan data interaksi Item menyimpan data historis dan waktu nyata dari interaksi antara pengguna dan item di katalog VIDEO_ON_DEMAND Anda. Untuk informasi selengkapnya tentang jenis data yang dapat Anda simpan dalam kumpulan data interaksi, lihatData interaksi item.

Anda harus memiliki kumpulan data interaksi Item untuk semua kasus penggunaan dan skema Anda harus memiliki bidang berikut:

- USER_ID () `string`
- ITEM_ID `string`
- STEMPEL WAKTU () `long`
- EVENT_TYPE (`string`dan tergantung pada kasus penggunaan, `Watch` dan `Click` jenis acara)

Skema Anda juga dapat menyertakan kata kunci cadangan berikut:

- EVENT_VALUE (,) `float null`
- KESAN (`string`,`null`)
- REKOMENDASI_ID (,) `string null`

Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda. Untuk contoh skema default untuk kumpulan data interaksi Item untuk domain VIDEO_ON_DEMAND, lihat. Skema Interaksi Default (domain VIDEO_ON_DEMAND)

Secara opsional tambahkan kata kunci cadangan EVENT_VALUE jika Anda memiliki data nilai untuk acara, seperti persentase video yang ditonton. Secara opsional tambahkan kata kunci cadangan IMPRESSION jika Anda ingin menyertakan data tayangan eksplisit dan implisit. Untuk informasi selengkapnya tentang merekam data tayangan, lihatData tayangan.

Jumlah total maksimum bidang metadata opsional yang dapat Anda tambahkan ke kumpulan data interaksi Item, dikombinasikan dengan jumlah total jenis peristiwa berbeda dalam data interaksi Item Anda, adalah 10. Bidang metadata yang disertakan dalam hitungan ini adalah bidang EVENT_TYPE, EVENT_VALUE bersama dengan bidang metadata kustom yang Anda tambahkan ke skema Anda. Jumlah maksimum bidang metadata yang tidak termasuk bidang cadangan, seperti IMPRESSION, adalah 5. Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki interaksi dengan nilai kategoris dengan lebih dari 1000, pekerjaan impor dataset Anda akan gagal.

Untuk informasi selengkapnya tentang persyaratan minimum dan batas data maksimum untuk kumpulan data interaksi Item untuk domain VIDEO_ON_DEMAND, lihat. Kuota layanan Untuk informasi tentang persyaratan skema Amazon Personalisasi umum, seperti persyaratan pemformatan dan tipe data bidang yang tersedia, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema, terlepas dari domain.

Skema Interaksi Default (domain VIDEO_ON_DEMAND)

Berikut ini adalah skema domain VIDEO_ON_DEMAND default untuk kumpulan data interaksi Item.

```
{

  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
```

```
            "name": "ITEM_ID",
            "type": "string"
        },
        {
            "name": "EVENT_TYPE",
            "type": "string"
        },
        {
            "name": "TIMESTAMP",
            "type": "long"
        }
    ],
    "version": "1.0"
}
```

Persyaratan kumpulan data pengguna (domain VIDEO_ON_DEMAND)

Dataset Pengguna menyimpan metadata tentang pengguna Anda. Ini mungkin termasuk informasi seperti usia, minat, jenis kelamin, dan keanggotaan loyalitas untuk setiap pengguna. Untuk informasi tentang jenis data pengguna yang dapat Anda impor ke Amazon Personalize, lihat. Metadata pengguna Untuk informasi tentang persyaratan skema Amazon Personalize umum, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema, terlepas dari domain.

Dataset Pengguna bersifat opsional untuk semua kasus penggunaan VIDEO_ON_DEMAND. Jika Anda memiliki data pengguna, kami sarankan untuk membuatnya untuk mendapatkan rekomendasi yang paling relevan. Jika Anda membuat kumpulan data Pengguna, skema Anda harus menyertakan bidang berikut.

- USER_ID
- 1 bidang metadata (kategoris `string` atau numerik)

Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda. Untuk contoh skema default untuk kumpulan data Pengguna untuk domain VIDEO_ON_DEMAND, lihat. Skema Pengguna Default (domain VIDEO_ON_DEMAND)

`SUBSCRIPTION_MODEL`Bidang disertakan dalam skema default. Bidang ini adalah kata kunci cadangan opsional dan harus memiliki jenis `string` dengan set kategoris ke. `true` Untuk

mendapatkan rekomendasi terbaik, kami menyarankan agar Anda menyimpan bidang ini dalam skema Anda jika Anda memiliki informasi model langganan tentang setiap pengguna Anda dalam data Anda. Data yang Anda impor harus sesuai dengan skema Anda.

Menggunakan data kategoris

Untuk menggunakan data kategoris, tambahkan bidang tipe `string` dan setel atribut kategoris bidang ke `true` dalam skema Anda. Kemudian sertakan data kategoris dalam file CSV massal Anda dan impor catatan individual. Untuk pengguna dengan beberapa kategori, pisahkan setiap nilai menggunakan bilah vertikal, '|'. Misalnya, untuk bidang SUBSCRIPTION_MODEL, data Anda untuk pengguna mungkin student|monthly|discount.

Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki pengguna dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Skema Pengguna Default (domain VIDEO_ON_DEMAND)

Berikut ini adalah skema domain VIDEO_ON_DEMAND default untuk kumpulan data Pengguna.

```
{

  "type": "record",
  "name": "Users",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "SUBSCRIPTION_MODEL",
          "type": "string",
          "categorical": true
      }
  ],
  "version": "1.0"
}
```

Persyaratan kumpulan data item (domain VIDEO_ON_DEMAND)

Dataset Item menyimpan metadata tentang item Anda di katalog Anda. Ini mungkin termasuk informasi seperti harga, genre, dan ketersediaan untuk setiap item. Untuk informasi tentang jenis

data item yang dapat Anda impor ke Amazon Personalize, lihat. Metadata barang Untuk informasi tentang persyaratan skema Amazon Personalisasi umum, seperti persyaratan pemformatan dan tipe data bidang yang tersedia, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema, terlepas dari domain.

Dataset Item diperlukan untuk beberapa kasus penggunaan (lihatKasus penggunaan VIDEO_ON_DEMAND). Saat opsional, kami tetap menyarankan untuk membuatnya untuk mendapatkan rekomendasi yang paling relevan. Jika Anda membuat kumpulan data Item, skema Anda harus menyertakan bidang berikut:

- ITEM_ID
- GENRE (kategoris`string`)
- CREATION_TIMESTAMP (dalam format waktu zaman Unix)

Skema Anda juga dapat menyertakan kata kunci cadangan berikut. Setiap kata kunci mencantumkan tipe data yang diperlukan dan apakah itu mendukung data nol. Menambahkan tipe null adalah opsional.

- HARGA (float)
- DURASI (mengapung)
- GENRE_L2 (kategoris,) `string` `null`
- GENRE_L3 (kategoris,) `string` `null`
- RATING_RATA-RATA (,) `float` `null`
- PRODUCT_DESCRIPTION (tekstual,) `string` `null`
- CONTENT_OWNER (kategoris`string`,`null`): Perusahaan yang memiliki video. Misalnya, nilai mungkin HBO, Paramount, dan NBC.
- CONTENT_CLASSIFICATION (kategoris`string`,`null`): Peringkat konten. Misalnya, nilai mungkin G, PG, PG-13, R, NC-17, dan tidak dinilai.

Untuk mendapatkan rekomendasi terbaik, kami sarankan Anda menyimpan ini sebanyak mungkin bidang ini dalam skema Anda karena Anda memiliki data. Data yang Anda impor harus sesuai dengan skema Anda. Jumlah maksimum kolom metadata adalah 100. Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda.

Gunakan kata kunci yang dicadangkan GENRE_L2 dan GENRE_L3 untuk item dengan beberapa kategori multi-level. Untuk informasi selengkapnya, lihat Menggunakan data kategoris. Untuk informasi tentang metadata tekstual dan kategoris lihat. Mempersiapkan metadata item untuk pelatihan Untuk contoh skema default untuk kumpulan data Item untuk domain ECOMMERCE, lihat. Skema Item default (domain VIDEO_ON_DEMAND)

Menggunakan data kategoris

Untuk menggunakan data kategoris, tambahkan bidang tipe `string` dan setel atribut kategoris bidang ke `true` dalam skema Anda. Kemudian sertakan data kategoris dalam file CSV massal Anda dan impor item individual. Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki item dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Untuk item dengan beberapa kategori, pisahkan setiap nilai dengan bilah vertikal, '|'. Misalnya, untuk bidang GENRES, data Anda untuk suatu item mungkin`Action|Crime|Biopic`. Jika Anda memiliki beberapa level data kategoris dan beberapa item memiliki beberapa kategori untuk setiap level dalam hierarki, tambahkan bidang untuk setiap level dan tambahkan indikator level setelah setiap nama bidang: GENRES, GENRE_L2, GENRE_L3. Ini memungkinkan Anda memfilter rekomendasi berdasarkan sub-kategori, bahkan jika item termasuk dalam beberapa kategori multi-level. Misalnya, video mungkin memiliki data berikut untuk setiap tingkat kategori:

- GENRE: Aksi|Petualangan
- GENRE_L2: Kejahatan|Barat
- GENRE_L3: biopik

Dalam contoh ini, video berada dalam hierarki aksi> kriminal> biopik dan hierarki biopik petualangan> barat>. Kami merekomendasikan hanya menggunakan hingga L3 tetapi Anda dapat menggunakan lebih banyak level jika perlu. Untuk informasi tentang membuat dan menggunakan filter, lihatMemfilter rekomendasi dan segmen pengguna.

Skema Item default (domain VIDEO_ON_DEMAND)

Berikut ini adalah skema default untuk kumpulan data Item untuk domain VIDEO_ON_DEMAND.

```
{
  "type": "record",
  "name": "Items",
```

```
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
    {
      "name": "ITEM_ID",
      "type": "string"
    },
    {
      "name": "GENRES",
      "type": [
        "string"
      ],
      "categorical": true
    },
    {
      "name": "CREATION_TIMESTAMP",
      "type": "long"
    }
  ],
  "version": "1.0"
}
```

Kumpulan data dan skema ECOMMERCE

Saat Anda membuat grup kumpulan data Domain untuk domain ECOMMERCE, setiap jenis kumpulan data memiliki skema default dengan sekumpulan bidang wajib dan direkomendasikan khusus eCommerce. Anda dapat menggunakan skema default atau membuat yang baru berdasarkan skema default. Data yang Anda impor harus sesuai dengan skema Anda dalam format dan jenis. Gunakan skema domain default yang tercantum di bagian di bawah ini sebagai panduan untuk menentukan data apa yang akan diimpor untuk membuat pemberi rekomendasi berbasis E-commerce Anda.

Anda bebas menambahkan bidang tambahan. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda.

Untuk informasi tentang persyaratan skema Amazon Personalisasi umum, seperti persyaratan pemformatan dan tipe data bidang yang tersedia, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema, terlepas dari domain.

Topik berikut memberikan informasi tentang bidang wajib dan direkomendasikan setiap kumpulan data untuk domain ECOMMERCE. Setiap bagian dataset menyertakan skema ECOMMERCE default dalam format JSON.

Topik

- Dataset domain ECOMMERCE dan persyaratan skema
- Persyaratan kumpulan data interaksi item (domain ECOMMERCE)
- Persyaratan kumpulan data pengguna (domain ECOMMERCE)
- Persyaratan kumpulan data item (domain ECOMMERCE)

Dataset domain ECOMMERCE dan persyaratan skema

Setiap jenis dataset memiliki bidang wajib berikut dan kata kunci yang dicadangkan. Kata kunci yang dicadangkan adalah opsional, bidang non-metadata. Bidang ini dianggap dicadangkan karena Anda harus menentukan bidang sebagai tipe data yang diperlukan saat Anda menggunakannya. Bidang string kategoris yang dicadangkan harus `categorical` disetel `ketrue`, sementara bidang string yang dicadangkan tidak dapat dikategorikan. Kata kunci tidak bisa ada dalam data Anda.

Jenis dataset	Bidang yang diperlukan	Kata kunci terpesan
Interaksi item (skema default)	USER_ID () `string` ITEM_ID () `string` STEMPEL WAKTU () `long` EVENT_TYPE (`string`dan tergantung pada kasus penggunaan, `Purchase` dan `View` jenis acara)	EVENT_VALUE (,) `float null` KESAN (`string`,`null`) REKOMENDASI_ID (,) `string null` EVENT_ATTRIBUTION_ SOURCE (,) `string null`
Pengguna (skema default)	USER_ID () `string` 1 bidang metadata (kategoris `string` atau numerik)	
Item (skema default)	ITEM_ID () `string` HARGA (`float`) CATEGORY_L1 (kategoris) `string`	KATEGORI_L2 (kategoris,) `string null` CATEGORY_L3 (kategoris,) `string null`

Jenis dataset	Bidang yang diperlukan	Kata kunci terpesan
		PRODUCT_DESCRIPTION (tekstual,) `string` `null` CREATION_TIMESTAMP () `long` AGE_GROUP (kategori `sstring`,) `null` DEWASA (kategori`sstring`,) `null` GENDER (kategori`sstring`,) `null`

Persyaratan kumpulan data interaksi item (domain ECOMMERCE)

Kumpulan data interaksi Item menyimpan data historis dan waktu nyata dari interaksi antara pengguna dan item di katalog ECOMMERCE Anda. Untuk informasi selengkapnya tentang jenis data yang dapat Anda simpan dalam kumpulan data interaksi, lihatData interaksi item. Untuk informasi tentang persyaratan skema Amazon Personalisasi umum, seperti persyaratan pemformatan dan tipe data bidang yang tersedia, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema, terlepas dari domain.

Anda harus setidaknya membuat kumpulan data interaksi Item dan skema Anda harus memiliki bidang berikut:

- USER_ID () `string`
- ITEM_ID () `string`
- STEMPEL WAKTU () `long`
- EVENT_TYPE (`string`dan tergantung pada kasus penggunaan, `Purchase` dan `View` jenis acara)

Skema Anda juga dapat menyertakan kata kunci cadangan berikut:

- EVENT_VALUE (,) `float` `null`

- KESAN (`string`,`null`)
- REKOMENDASI_ID (,) `string` `null`

Data yang Anda impor harus sesuai dengan skema Anda. Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda. Untuk contoh skema default untuk kumpulan data interaksi Item untuk domain ECOMMERCE, lihat. Skema Interaksi Default (domain ECOMMERCE)

Secara opsional tambahkan kata kunci cadangan EVENT_VALUE jika Anda memiliki data nilai untuk acara. Secara opsional tambahkan kata kunci cadangan IMPRESSION jika Anda ingin menyertakan data tayangan eksplisit dan implisit. Untuk informasi selengkapnya tentang merekam data tayangan, lihatData tayangan.

Jumlah total maksimum bidang metadata opsional yang dapat Anda tambahkan ke kumpulan data interaksi Item, dikombinasikan dengan jumlah total jenis peristiwa berbeda dalam data interaksi Item Anda, adalah 10. Bidang metadata yang disertakan dalam hitungan ini adalah bidang EVENT_TYPE, EVENT_VALUE bersama dengan bidang metadata kustom yang Anda tambahkan ke skema Anda. Jumlah maksimum bidang metadata yang tidak termasuk bidang cadangan, seperti IMPRESSION, adalah 5. Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki interaksi dengan nilai kategoris dengan lebih dari 1000, pekerjaan impor dataset Anda akan gagal.

Untuk informasi selengkapnya tentang persyaratan minimum dan batas data maksimum untuk kumpulan data interaksi Item untuk domain ECOMMERCE, lihat. Kuota layanan

Skema Interaksi Default (domain ECOMMERCE)

Berikut ini adalah skema domain ECOMMERCE default untuk kumpulan data interaksi Item.

```
{

  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
```

```
      {
          "name": "ITEM_ID",
          "type": "string"
      },
      {
          "name": "EVENT_TYPE",
          "type": "string"
      },
      {
          "name": "TIMESTAMP",
          "type": "long"
      }
  ],
  "version": "1.0"
}
```

Persyaratan kumpulan data pengguna (domain ECOMMERCE)

Dataset Pengguna menyimpan metadata tentang pengguna Anda. Ini mungkin termasuk informasi seperti usia, jenis kelamin, dan keanggotaan loyalitas untuk setiap pengguna. Untuk informasi selengkapnya tentang jenis data pengguna yang dapat Anda impor ke Amazon Personalize, lihat. Metadata pengguna Untuk informasi tentang persyaratan skema Amazon Personalisasi umum, seperti persyaratan pemformatan dan tipe data bidang yang tersedia, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema, terlepas dari domain.

Dataset Pengguna adalah opsional untuk semua kasus penggunaan ECOMMERCE. Jika Anda memiliki data pengguna, kami sarankan untuk membuatnya untuk mendapatkan rekomendasi yang paling relevan. Jika Anda membuat kumpulan data Pengguna, skema Anda harus menyertakan bidang berikut.

- USER_ID
- 1 bidang metadata (kategoris `string` atau numerik)

Data yang Anda impor harus sesuai dengan skema Anda. Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda. Untuk contoh skema default untuk kumpulan data Pengguna untuk domain ECOMMERCE, lihat. Skema Pengguna Default (domain ECOMMERCE)

Untuk informasi selengkapnya tentang persyaratan minimum dan batas data maksimum untuk kumpulan data Pengguna, lihatKuota layanan.

Menggunakan data kategoris

Untuk menggunakan data kategoris, tambahkan bidang tipe `string` dan setel atribut kategoris bidang ke `true` dalam skema Anda. Kemudian sertakan data kategoris dalam file CSV massal Anda dan impor catatan individual. Untuk pengguna dengan beberapa kategori, pisahkan setiap nilai menggunakan bilah vertikal, '|'. Misalnya, untuk bidang SUBSCRIPTION_MODEL, data Anda untuk pengguna mungkin student|monthly|discount.

Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki pengguna dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Skema Pengguna Default (domain ECOMMERCE)

Berikut ini adalah skema domain ECOMMERCE default untuk kumpulan data Pengguna dengan bidang KATEGORI sebagai bidang metadata yang diperlukan.

```
{

  "type": "record",
  "name": "Users",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "MEMBERSHIP_STATUS",
          "type": "string",
          "categorical": true
      }
  ],
  "version": "1.0"
}
```

Persyaratan kumpulan data item (domain ECOMMERCE)

Dataset Item menyimpan metadata tentang item ECOMMERCE Anda. Ini mungkin termasuk informasi seperti harga, kategori, dan deskripsi produk untuk setiap item. Untuk informasi

selengkapnya tentang jenis data item yang dapat Anda impor ke Amazon Personalize, lihat. Metadata barang Untuk informasi tentang persyaratan skema Amazon Personalisasi umum, seperti persyaratan pemformatan dan tipe data bidang yang tersedia, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema, terlepas dari domain.

Dataset Item adalah opsional untuk semua kasus penggunaan ECOMMERCE. Jika Anda memiliki data item, sebaiknya buat satu untuk mendapatkan rekomendasi yang paling relevan. Jika Anda membuat kumpulan data item, skema Anda harus menyertakan bidang berikut:

- ITEM_ID
- HARGA (`float`)
- CATEGORY_L1 (kategoris) `string`

Skema Anda juga dapat menyertakan kata kunci cadangan berikut. Untuk bidang kategoris, Anda dapat menentukan rentang nilai Anda sendiri berdasarkan kasus penggunaan Anda.

- KATEGORI_L2 (kategoris,) `string` `null`
- CATEGORY_L3 (kategoris,) `string` `null`
- PRODUCT_DESCRIPTION (tekstual,) `string` `null`
- CREATION_TIMESTAMP () `float`
- AGE_GROUP (kategoris`string`,`null`): Untuk kelompok usia item tersebut. Nilai mungkin bayi baru lahir, bayi, anak-anak, dan orang dewasa.
- DEWASA (kategoris`string`,`null`): Apakah barang dibatasi hanya untuk orang dewasa, seperti alkohol. Nilai mungkin ya atau tidak.
- GENDER (kategoris`string`,`null`): Jenis kelamin item tersebut. Nilai mungkin laki-laki, perempuan, dan unisex.

Untuk mendapatkan rekomendasi terbaik, kami sarankan Anda menyimpan ini sebanyak mungkin bidang ini dalam skema Anda karena Anda memiliki data. Data yang Anda impor harus sesuai dengan skema Anda. Data yang Anda impor harus sesuai dengan skema Anda. Jumlah maksimum kolom metadata adalah 100. Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda.

Gunakan kata kunci yang dicadangkan CATEGORY_L2 dan CATEGORY_L3 untuk item dengan beberapa kategori multi-level. Untuk informasi selengkapnya, lihat Menggunakan data kategoris.

Untuk informasi tentang metadata tekstual dan kategoris lihat. Metadata teks tidak terstruktur Untuk contoh skema default untuk kumpulan data Item untuk domain ECOMMERCE, lihat. Skema Item Default (domain ECOMMERCE)

Menggunakan data kategoris

Untuk menggunakan data kategoris, tambahkan bidang tipe `string` dan setel atribut kategoris bidang ke `true` dalam skema Anda. Kemudian sertakan data kategoris dalam file CSV massal Anda dan impor item individual. Anda dapat menentukan rentang nilai Anda sendiri berdasarkan kasus penggunaan Anda. Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki item dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Untuk item dengan beberapa kategori, pisahkan setiap nilai dengan bilah vertikal, '|'. Misalnya, untuk bidang CATEGORY_L1, data Anda untuk suatu item mungkin. `Electronics|Productivity|Mouse` Jika Anda memiliki beberapa level data kategoris dan beberapa item memiliki beberapa kategori untuk setiap level dalam hierarki, tambahkan bidang untuk setiap level dan tambahkan indikator level setelah setiap nama bidang: CATEGORY_L1, CATEGORY_L2, CATEGORY_L3. Ini memungkinkan Anda memfilter rekomendasi berdasarkan sub-kategori, bahkan jika item termasuk dalam beberapa kategori multi-level. Misalnya, item mungkin memiliki data berikut untuk setiap tingkat kategori:

- KATEGORI_L1: Elektronik|Produktivitas
- KATEGORI_L2: Produktivitas|Komputer
- KATEGORI_L3: Mouse

Dalam contoh ini, item tersebut ada dalam hierarki elektronik> produktivitas> mouse dan hierarki produktivitas > komputer > mouse. Kami merekomendasikan hanya menggunakan hingga L3 tetapi Anda dapat menggunakan lebih banyak level jika perlu. Untuk informasi tentang membuat dan menggunakan filter lihatMemfilter rekomendasi dan segmen pengguna.

Skema Item Default (domain ECOMMERCE)

Berikut ini adalah skema default untuk kumpulan data Item untuk domain ECOMMERCE dengan hanya bidang yang diperlukan.

```
{
  "type": "record",
```

```
  "name": "Items",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
    {
      "name": "ITEM_ID",
      "type": "string"
    },
    {
      "name": "PRICE",
      "type": "float"
    },
    {
      "name": "CATEGORY_L1",
      "type": [
        "string"
      ],
      "categorical": true
    }
  ],
  "version": "1.0"
}
```

Kumpulan data dan skema khusus

Saat membuat grup kumpulan data Kustom, Anda membuat skema sendiri dari awal. Kumpulan data dan skema grup kumpulan data khusus memiliki lebih sedikit bidang yang diperlukan dan lebih banyak fleksibilitas. Topik berikut menjelaskan skema dan persyaratan data untuk kumpulan data grup kumpulan data Kustom. Setiap bagian kumpulan data mencantumkan data yang diperlukan untuk jenis kumpulan data dan memberikan contoh skema JSON.

Untuk informasi tentang jenis data yang dapat Anda impor ke Amazon Personalize, lihat. Jenis data yang dapat digunakan Amazon Personalize Untuk informasi tentang persyaratan skema Amazon Personalisasi umum, seperti persyaratan pemformatan dan tipe data bidang yang tersedia, lihat. Membuat file JSON skema untuk skema Amazon Personalize Persyaratan ini berlaku untuk semua skema Amazon Personalize.

Topik

- Dataset kustom dan persyaratan skema
- Persyaratan skema kumpulan data interaksi item (khusus)
- Persyaratan skema kumpulan data pengguna (kustom)

- Persyaratan skema dataset item (kustom)
- Persyaratan skema kumpulan data tindakan (kustom)
- Persyaratan skema kumpulan data interaksi tindakan (kustom)

Dataset kustom dan persyaratan skema

Saat Anda membuat kumpulan data untuk grup kumpulan data Kustom, setiap jenis kumpulan data memiliki bidang wajib berikut dan kata kunci cadangan dengan tipe data yang diperlukan.

Jenis dataset	Bidang wajib	Kata kunci terpesan
Interaksi item (contoh skema)	USER_ID () `string` ITEM_ID () `string` STEMPEL WAKTU () `long`	EVENT_TYPE () `string` EVENT_VALUE (,) `float null` KESAN (`string`,`null`) REKOMENDASI_ID (,) `string null` EVENT_ATTRIBUTION_SOURCE (,) `string null`
Pengguna (contoh skema)	USER_ID () `string` 1 bidang metadata (kategoris `string` atau numerik)	
Item (contoh skema)	ITEM_ID () `string` 1 bidang metadata (bidang kategoris atau `string` tekstual atau bidang numerik)	CREATION_TIMESTAMP () `long`
Tindakan (contoh skema)	ACTION_ID () `string` 1 bidang metadata (kategoris `string` atau numerik)	CREATION_TIMESTAMP () `long` NILAI (`long`,`null`)

Jenis dataset	Bidang wajib	Kata kunci terpesan
		JENIS (`string`,`null`) EXPIRATION_TIMESTAMP (,) `long` `null` REPEAT_FREKUENSI (`long`,) `null`
Interaksi aksi (contoh skema)	USER_ID () `string` ACTION_ID () `string` EVENT_TYPE () `string` STEMPEL WAKTU () `long`	KESAN (`string`,`null`) REKOMENDASI_ID (,) `string` `null`

Kolom metadata

Metadata mencakup bidang string atau non-string yang tidak diperlukan atau tidak menggunakan kata kunci cadangan. Skema metadata memiliki batasan berikut:

- Skema Pengguna, Item, dan Tindakan memerlukan setidaknya satu bidang metadata.
- Anda dapat menambahkan paling banyak 25 bidang metadata untuk skema Pengguna, 100 bidang metadata untuk skema Item, dan 10 bidang metadata untuk skema Tindakan.
- Jika Anda menambahkan bidang metadata Anda sendiri dari tipe`string`, itu harus menyertakan `categorical` atribut atau `textual` atribut (hanya skema Item yang mendukung bidang dengan atribut tekstual). Jika tidak, Amazon Personalize tidak akan menggunakan bidang saat melatih model.

Kata kunci terpesan

Kata kunci yang dicadangkan adalah opsional, bidang non-metadata. Bidang ini dianggap dicadangkan karena Anda harus menentukan bidang sebagai tipe data yang diperlukan saat Anda menggunakannya, dan kata kunci tidak dapat digunakan sebagai nilai dalam data Anda. Bidang string kategoris yang dicadangkan harus `categorical` disetel `ketrue`, sementara bidang string yang dicadangkan tidak dapat dikategorikan. Berikut ini adalah kata kunci yang dicadangkan:

- EVENT_TYPE: Untuk kumpulan data interaksi Item dengan satu atau beberapa jenis acara, seperti klik dan unduh, gunakan bidang. `EVENT_TYPE` Anda harus mendefinisikan bidang EVENT_TYPE sebagai `string` dan tidak dapat diatur sebagai kategoris.
- EVENT_VALUE: Untuk kumpulan data interaksi Item yang menyertakan data nilai untuk peristiwa, seperti persentase video yang ditonton pengguna, gunakan `EVENT_VALUE` bidang dengan tipe dan opsional. `float` `null`
- CREATION_TIMESTAMP: Untuk kumpulan data Item atau Tindakan dengan stempel waktu untuk tanggal pembuatan setiap item, gunakan bidang dengan tipe. `CREATION_TIMESTAMP` `long` Amazon Personalize menggunakan `CREATION_TIMESTAMP` data untuk menghitung usia item dan menyesuaikan rekomendasi yang sesuai. Lihat Data stempel waktu pembuatan.
- IMPRESSION: Untuk kumpulan data interaksi Item dengan data tayangan eksplisit, gunakan `IMPRESSION` bidang dengan tipe `String` dan tipe opsional. `null` Tayangan adalah daftar item yang terlihat oleh pengguna saat berinteraksi dengan (misalnya, diklik atau ditonton) item tertentu. Untuk informasi selengkapnya, lihat Data tayangan.
- RECOMMENTATION_ID: Untuk kumpulan data interaksi Item yang menggunakan rekomendasi sebelumnya sebagai data tayangan implisit, secara opsional gunakan bidang dengan tipe dan tipe opsional. `RECOMMENDATION_ID` `String` `null`

  Anda tidak perlu menambahkan `RECOMMENDATION_ID` bidang untuk Amazon Personalize untuk menggunakan tayangan implisit saat membuat rekomendasi. Anda dapat lulus `recommendationId` dalam PutEvents operasi tanpa itu. Untuk informasi selengkapnya, lihat Data tayangan.
- NILAI: Untuk kumpulan data Tindakan, jika Anda memiliki nilai data untuk beberapa atau semua tindakan Anda, tambahkan `VALUE` bidang ke skema Anda. Untuk jenisnya, gunakan `long` dan ketik `null` opsional. Untuk informasi selengkapnya tentang tindakan dan nilainya, lihatData nilai.
- ACTION_EXPIRATION_TIMESTAMP: Untuk kumpulan data Tindakan, jika Anda memiliki stempel waktu kedaluwarsa untuk beberapa atau semua tindakan Anda, tambahkan bidang ke skema Anda. `ACTION_EXPIRATION_TIMESTAMP` Untuk jenisnya, gunakan `long` dan ketik `null` opsional. Untuk informasi selengkapnya tentang cap waktu kedaluwarsa, lihat. Data stempel waktu kedaluwarsa tindakan
- REPEAT_FREQUENCY: Untuk kumpulan data Tindakan, jika Anda memiliki data frekuensi berulang untuk beberapa atau semua tindakan Anda, tambahkan `REPEAT_FREQUENCY` bidang ke skema Anda. Untuk jenisnya, gunakan `long` dan ketik `null` opsional. Untuk informasi selengkapnya tentang data frekuensi berulang, lihatUlangi data frekuensi.

Persyaratan skema kumpulan data interaksi item (khusus)

Kumpulan data interaksi Item menyimpan data historis dan waktu nyata dari interaksi antara pengguna dan item di katalog Anda. Untuk informasi tentang jenis data interaksi yang dapat digunakan Amazon Personalize, lihat. Data interaksi item

Data yang Anda berikan untuk setiap interaksi harus sesuai dengan skema Anda. Bergantung pada skema Anda, metadata interaksi dapat menyertakan nilai kosong/nol. Minimal, Anda harus memberikan yang berikut untuk setiap interaksi:

- ID Pengguna
- ID Barang
- Stempel waktu (dalam format waktu zaman Unix)

Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda.

Jumlah total maksimum bidang metadata opsional yang dapat Anda tambahkan ke kumpulan data interaksi Item, dikombinasikan dengan jumlah total jenis peristiwa berbeda dalam data interaksi Item Anda, adalah 10. Bidang metadata yang disertakan dalam hitungan ini adalah bidang EVENT_TYPE, EVENT_VALUE bersama dengan bidang metadata kustom yang Anda tambahkan ke skema Anda. Jumlah maksimum bidang metadata yang tidak termasuk bidang cadangan, seperti IMPRESSION, adalah 5. Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki interaksi dengan nilai kategoris dengan lebih dari 1000, pekerjaan impor dataset Anda akan gagal.

Untuk informasi lebih lanjut tentang persyaratan minimum dan batas data maksimum untuk kumpulan data interaksi Item, lihatKuota layanan.

Contoh skema interaksi (kustom)

Contoh berikut menunjukkan skema untuk kumpulan data interaksi Item. `TIMESTAMP`Bidang `USER_ID`dan `ITEM_ID`, dan diperlukan. `IMPRESSION`Bidang`EVENT_TYPE`,`EVENT_VALUE`, dan adalah kata kunci cadangan opsional yang diakui oleh Amazon Personalize. EVENT_TYPE harus bertipe string dan tidak bisa kategoris. `LOCATION`dan `DEVICE` merupakan bidang metadata kontekstual opsional. Untuk informasi tentang persyaratan skema, lihatDataset kustom dan persyaratan skema.

```
{
```

```
  "type": "record",
  "name": "Interactions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "ITEM_ID",
          "type": "string"
      },
      {
          "name": "EVENT_TYPE",
          "type": "string"
      },
      {
          "name": "EVENT_VALUE",
          "type": [
             "float",
             "null"
          ]
      },
      {
          "name": "LOCATION",
          "type": "string",
          "categorical": true
      },
      {
          "name": "DEVICE",
          "type": [
              "string",
              "null"
          ],
          "categorical": true
      },
      {
          "name": "TIMESTAMP",
          "type": "long"
      },
      {
          "name": "IMPRESSION",
          "type": "string"
      }
```

```
  ],
  "version": "1.0"
}
```

Untuk skema ini, beberapa baris pertama data historis dalam file CSV mungkin terlihat seperti berikut. Perhatikan bahwa beberapa nilai untuk EVENT_VALUE adalah null.

```
USER_ID,ITEM_ID,EVENT_TYPE,EVENT_VALUE,LOCATION,DEVICE,TIMESTAMP,IMPRESSION
35,73,click,,Ohio,Tablet,1586731606,73|70|17|95|96|92|55|45|16|97|56|54|33|94|36|10|5|
43|19|13|51|90|65|59|38
54,35,watch,0.75,Indiana,Cellphone,1586735164,35|82|78|57|20|63|1|90|76|75|49|71|26|24|
25|6|37|85|40|98|32|13|11|54|48
9,33,click,,Oregon,Cellphone,1586735158,68|33|62|6|15|57|45|24|78|89|90|40|26|91|66|31|
47|17|99|29|27|41|77|75|14
23,10,watch,0.25,California,Tablet,1586735697,92|89|36|10|39|77|4|27|79|18|83|16|28|68|
78|40|50|3|99|7|87|49|12|57|53
27,11,watch,0.55,Indiana,Tablet,1586735763,11|7|39|95|71|1|6|40|41|28|99|53|68|76|0|65|
69|36|22|42|34|67|24|20|66
...
...
```

Persyaratan skema kumpulan data pengguna (kustom)

Dataset Pengguna menyimpan metadata tentang pengguna Anda. Ini mungkin termasuk informasi seperti usia, jenis kelamin, dan keanggotaan loyalitas untuk setiap item. Untuk informasi tentang jenis data pengguna yang dapat Anda impor ke Amazon Personalize, lihat. Metadata pengguna

Data yang Anda berikan untuk setiap pengguna harus sesuai dengan skema Anda. Minimal, Anda harus memberikan User ID untuk setiap pengguna (panjang maksimal 256 karakter). Bergantung pada skema Anda, metadata pengguna dapat menyertakan nilai. empty/null Skema Pengguna Anda harus memiliki minimal satu bidang metadata, tetapi jika Anda menambahkan `null` tipe, nilai ini dapat menjadi nol untuk pengguna. Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda.

Untuk menggunakan data kategoris, tambahkan bidang tipe `string` dan atur atribut kategoris bidang ke `true` dalam skema Anda. Kemudian sertakan data kategoris dalam file CSV massal Anda dan impor catatan individual. Untuk pengguna dengan beberapa kategori, pisahkan setiap nilai menggunakan bilah vertikal, '|'. Misalnya, untuk bidang SUBSCRIPTION_MODEL, data Anda untuk pengguna mungkin student|monthly|discount.

Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki pengguna dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Untuk informasi selengkapnya tentang persyaratan minimum dan batas data maksimum untuk kumpulan data Pengguna, lihatKuota layanan.

Contoh skema pengguna (kustom)

Contoh berikut menunjukkan bagaimana menyusun skema Users. `USER_ID`Bidang diperlukan dan `GENDER` bidang `AGE` dan adalah metadata. Setidaknya satu bidang metadata diperlukan dan Anda dapat menambahkan paling banyak 25 bidang metadata. Untuk informasi tentang persyaratan skema, lihatDataset kustom dan persyaratan skema.

```
{
  "type": "record",
  "name": "Users",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "AGE",
          "type": "int"
      },
      {
          "name": "GENDER",
          "type": "string",
          "categorical": true
      }
  ],
  "version": "1.0"
}
```

Untuk skema ini, beberapa baris pertama data historis dalam file CSV mungkin terlihat seperti berikut.

```
USER_ID,AGE,GENDER
5,34,Male
6,56,Female
8,65,Male
...
```

```
...
```

Persyaratan skema dataset item (kustom)

Dataset Item menyimpan metadata tentang item Anda di katalog Anda. Ini mungkin termasuk informasi seperti harga, genre, dan ketersediaan untuk setiap item. Untuk informasi tentang jenis data item yang dapat Anda impor ke Amazon Personalize, lihat. Metadata barang

Data yang Anda berikan untuk setiap item harus cocok dengan skema kumpulan data Item Anda. Minimal, Anda harus memberikan ID Item untuk setiap item (panjang maksimal 256 karakter). Bergantung pada skema Anda, metadata item dapat menyertakan nilai. empty/null Skema Anda harus memiliki minimal satu bidang metadata, tetapi jika Anda menambahkan `null` tipe, nilai ini dapat menjadi nol untuk item tersebut. Anda bebas menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Selama bidang tidak terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema, nama bidang dan tipe data terserah Anda.

Untuk menggunakan data kategoris, tambahkan bidang tipe `string` dan atur atribut kategoris bidang ke `true` dalam skema Anda. Kemudian sertakan data kategoris dalam file CSV massal Anda dan impor item individual. Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki item dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Untuk item dengan beberapa kategori, pisahkan setiap nilai dengan bilah vertikal, '|'. Misalnya, untuk bidang GENRES, data Anda untuk item mungkin`Action|Crime|Biopic`. Jika Anda memiliki beberapa level data kategoris dan beberapa item memiliki beberapa kategori untuk setiap level dalam hierarki, tambahkan bidang untuk setiap level dan tambahkan indikator level setelah setiap nama bidang: GENRES, GENRE_L2, GENRE_L3. Ini memungkinkan Anda memfilter rekomendasi berdasarkan sub-kategori, bahkan jika item termasuk dalam beberapa kategori multi-level (untuk informasi tentang membuat dan menggunakan filter lihatMemfilter rekomendasi dan segmen pengguna). Misalnya, video mungkin memiliki data berikut untuk setiap tingkat kategori:

- GENRE: Aksi|Petualangan
- GENRE_L2: Kejahatan|Barat
- GENRE_L3: Biopik

Dalam contoh ini, video berada dalam hierarki aksi> kriminal> biopik dan hierarki biopik petualangan> barat>. Kami merekomendasikan hanya menggunakan hingga L3 tetapi Anda dapat menggunakan lebih banyak level jika perlu.

Selama pelatihan model, Amazon Personalize mempertimbangkan maksimum 750.000 item. Jika Anda mengimpor lebih dari 750.000 item, Amazon Personalize memutuskan item mana yang akan disertakan dalam pelatihan, dengan penekanan pada menyertakan item baru (item yang baru-baru ini Anda tambahkan tanpa interaksi) dan item yang ada dengan data interaksi terbaru.

Untuk informasi selengkapnya tentang persyaratan minimum dan batas data maksimum untuk kumpulan data Item, lihatKuota layanan.

Contoh skema kumpulan data item (kustom)

Contoh berikut menunjukkan bagaimana menyusun skema Item. Bidang `ITEM_ID` wajib diisi. `GENRE`Bidang adalah metadata kategoris dan `DESCRIPTION` bidangnya adalah metadata tekstual. Setidaknya diperlukan satu bidang metadata. Anda dapat menambahkan maksimal 100 bidang metadata. `CREATION_TIMESTAMP`Bidang adalah kata kunci yang dicadangkan. Untuk informasi tentang persyaratan skema, lihatDataset kustom dan persyaratan skema.

```
{
  "type": "record",
  "name": "Items",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
    {
      "name": "ITEM_ID",
      "type": "string"
    },
    {
      "name": "GENRES",
      "type": [
        "null",
        "string"
      ],
      "categorical": true
    },
    {
      "name": "CREATION_TIMESTAMP",
      "type": "long"
    },
    {
      "name": "DESCRIPTION",
      "type": [
        "null",
        "string"
```

```
      ],
      "textual": true
    }
  ],
  "version": "1.0"
}
```

Untuk skema ini, beberapa baris pertama data historis dalam file CSV mungkin terlihat seperti berikut.

```
ITEM_ID,GENRES,CREATION_TIMESTAMP,DESCRIPTION
1,Adventure|Animation|Children|Comedy|Fantasy,1570003267,"This is an animated movie
 that features action, comedy, and fantasy. Audience is children. This movie was
 released in 2004."
2,Adventure|Children|Fantasy,1571730101,"This is an adventure movie with elements of
 fantasy. Audience is children. This movie was release in 2010."
3,Comedy|Romance,1560515629,"This is a romantic comedy. The movie was released in 1999.
 Audience is young women."
4,Comedy|Drama|Romance,1581670067,"This movie includes elements of both comedy and
 drama as well as romance. This movie was released in 2020."
...
...
```

Persyaratan skema kumpulan data tindakan (kustom)

Tindakan adalah aktivitas keterlibatan yang mungkin ingin Anda rekomendasikan kepada pelanggan Anda. Tindakan mungkin termasuk menginstal aplikasi seluler Anda, melengkapi profil keanggotaan, bergabung dengan program loyalitas Anda, atau mendaftar untuk email promosi. Dataset Actions menyimpan data tentang tindakan Anda. Untuk informasi tentang jenis data tindakan yang dapat Anda impor ke Amazon Personalize, lihat. Metadata aksi

Data yang Anda berikan untuk setiap tindakan harus sesuai dengan skema kumpulan data Tindakan Anda. Bergantung pada skema Anda, metadata tindakan dapat menyertakan nilai. empty/null

Minimal, Anda harus memberikan ID Tindakan untuk setiap item (panjang maksimal 256 karakter). Skema Anda harus memiliki minimal satu bidang metadata, tetapi jika Anda menambahkan `null` tipe, nilai ini dapat menjadi nol untuk tindakan tersebut. Anda dapat menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Anda dapat memilih nama bidang dan tipe data kecuali bidang terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema.

Untuk menambahkan bidang kategoris, tambahkan bidang tipe `string` dan atur atribut kategoris bidang ke `true` dalam skema Anda. Kemudian sertakan data kategoris dalam file CSV massal Anda dan impor tindakan individual. Nilai kategoris dapat memiliki paling banyak 1000 karakter. Jika Anda memiliki tindakan dengan nilai kategoris dengan lebih dari 1000 karakter, pekerjaan impor dataset Anda akan gagal.

Untuk informasi selengkapnya tentang persyaratan minimum dan batas data maksimum untuk kumpulan data Tindakan, lihatKuota layanan.

Contoh skema kumpulan data tindakan (kustom)

Contoh berikut menunjukkan bagaimana menyusun skema Actions. Bidang `ACTION_ID` wajib diisi. `MEMBERSHIP_LEVEL`Bidang adalah bidang string kategoris. `REPEAT_FREQUENCY`Bidang `VALUECREATION_TIMESTAMP`,, dan adalah kata kunci yang dicadangkan dengan jenis yang diperlukan. Anda dapat menambahkan maksimal 10 kolom. Untuk informasi tentang persyaratan skema, lihatDataset kustom dan persyaratan skema.

```
{
  "type": "record",
  "name": "Actions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
    {
      "name": "ACTION_ID",
      "type": "string"
    },
    {
      "name": "VALUE",
      "type": [
        "null",
        "long"
      ]
    },

    {
      "name": "MEMBERSHIP_LEVEL",
      "type": [
        "null",
        "string"
      ],
      "categorical": true
    },
```

```
    {
      "name": "CREATION_TIMESTAMP",
      "type": "long"
    },
    {
      "name": "REPEAT_FREQUENCY",
      "type": [
        "long",
        "null"
      ]
    }
  ],
  "version": "1.0"
}
```

Untuk skema ini, beberapa baris pertama data historis dalam file CSV mungkin terlihat seperti berikut.

```
ACTION_ID,VALUE,MEMBERSHIP_LEVEL,CREATION_TIMESTAMP,REPEAT_FREQUENCY
1,10,Deluxe|Premium,1510003267,7
2,5,Basic,1580003267,7
3,5,Preview,1590003267,3
4,10,Deluxe|Platinum,1560003267,4
...
...
```

Persyaratan skema kumpulan data interaksi tindakan (kustom)

Kumpulan data interaksi Tindakan menyimpan data historis dan waktu nyata dari interaksi antara pengguna dan tindakan dalam kumpulan data Tindakan Anda. Untuk informasi tentang jenis data yang dapat digunakan Amazon Personalize, lihat. Data interaksi aksi

Data yang Anda berikan untuk setiap interaksi harus sesuai dengan skema Anda. Bergantung pada skema Anda, metadata interaksi dapat menyertakan nilai. empty/null Minimal, skema Anda harus mencakup yang berikut:

- USER_ID
- ACTION_ID
- TIMESTAMP
- EVENT_TYPE

Anda dapat menambahkan bidang tambahan tergantung pada kasus penggunaan dan data Anda. Anda dapat memilih nama bidang dan tipe data kecuali bidang terdaftar sebagai wajib atau dicadangkan, dan tipe data dicantumkanJenis data skema.

Untuk informasi selengkapnya tentang persyaratan minimum dan batas data maksimum untuk kumpulan data interaksi Tindakan, lihatKuota layanan.

Contoh skema kumpulan data interaksi aksi (kustom)

Contoh berikut menunjukkan skema untuk kumpulan data interaksi Tindakan dengan hanya bidang yang diperlukan. Untuk informasi tentang persyaratan pemformatan skema umum, lihat. Persyaratan pemformatan skema

```
{

  "type": "record",
  "name": "ActionInteractions",
  "namespace": "com.amazonaws.personalize.schema",
  "fields": [
      {
          "name": "USER_ID",
          "type": "string"
      },
      {
          "name": "ACTION_ID",
          "type": "string"
      },
      {
          "name": "EVENT_TYPE",
          "type": "string"
      },
      {
          "name": "TIMESTAMP",
          "type": "long"
      }
  ],
  "version": "1.0"
}
```

Untuk skema ini, beberapa baris pertama data historis dalam file CSV mungkin terlihat seperti berikut.

```
USER_ID,ACTION_ID,EVENT_TYPE,TIMESTAMP
```

```
35,73,Viewed,1586731606
54,35,Not taken,1586731609
9,33,Viewed,1586735158
23,10,Taken,1586735697
27,11,Taken,1586735763
...
...
```

Daftar periksa kesiapan

Daftar periksa ini menyediakan daftar fitur, persyaratan, dan panduan data Amazon Personalize. Ini dapat membantu Anda merencanakan, atau Anda dapat menggunakannya sebagai referensi saat Anda membuat sumber daya di Amazon Personalisasi.

Topik

- Sudahkah Anda mencocokkan kasus penggunaan Anda dengan sumber daya Amazon Personalize?
- Apakah Anda memiliki data interaksi item yang cukup?
- Apakah Anda memiliki arsitektur streaming acara real-time?
- Apakah data Anda dioptimalkan untuk Amazon Personalize?
- Apakah Anda mengumpulkan data opsional yang dapat meningkatkan rekomendasi?
- Apakah Anda memiliki rencana untuk menguji rekomendasi Anda?
- Apakah Anda memiliki tujuan bisnis tambahan?

Sudahkah Anda mencocokkan kasus penggunaan Anda dengan sumber daya Amazon Personalize?

Rekomendasi Amazon Personalize dapat mengatasi kasus penggunaan berikut:

- Menghasilkan rekomendasi yang dipersonalisasi untuk pengguna
- Merekomendasikan item serupa atau terkait
- Merekomendasikan item yang sedang tren atau populer
- Merekomendasikan tindakan terbaik berikutnya untuk pengguna
- Mengurutkan ulang berdasarkan relevansi (hanya dengan sumber daya khusus)
- Menghasilkan segmen pengguna (hanya dengan sumber daya khusus)

Amazon Personalisasi menampilkan sumber daya berbasis domain dan sumber daya khusus yang dikonfigurasi untuk kasus penggunaan ini. Anda mulai dengan membuat grup dataset Domain atau grup kumpulan data Kustom:

- Dengan grup kumpulan data Domain, Anda membuat sumber daya yang telah dikonfigurasi sebelumnya dan dioptimalkan untuk domain VIDEO_ON_DEMAND atau ECOMMERCE.

  Jika Anda memiliki aplikasi streaming video atau e-commerce, kami sarankan Anda memulai dengan grup kumpulan data Domain. Anda masih dapat menambahkan sumber daya khusus, seperti solusi dan versi solusi yang dilatih untuk kasus penggunaan khusus. Dan Anda masih dapat menggunakan sumber daya khusus untuk mendapatkan rekomendasi batch. Anda tidak dapat membuat sumber daya tindakan terbaik berikutnya, termasuk kumpulan data Tindakan dan Interaksi Tindakan, dalam grup kumpulan data domain.
- Dengan grup kumpulan data Kustom, Anda memilih resep yang cocok dengan kasus penggunaan Anda. Anda kemudian melatih dan menerapkan hanya solusi dan versi solusi yang dapat dikonfigurasi (model rekomendasi Amazon Personalize terlatih). Saat siap, Anda dapat menerapkan versi solusi dalam kampanye untuk rekomendasi waktu nyata. Atau Anda bisa mendapatkan rekomendasi batch tanpa kampanye.

  Jika Anda tidak memiliki aplikasi streaming video atau e-commerce, kami sarankan Anda membuat grup dataset khusus. Jika tidak, mulailah dengan grup dataset Domain dan tambahkan sumber daya khusus seperlunya.

Untuk informasi tentang kasus penggunaan dan resep khusus yang tersedia di Amazon Personalize, lihat. Mencocokkan kasus penggunaan Anda dengan Amazon Personalize resource

Apakah Anda memiliki data interaksi item yang cukup?

Untuk semua kasus penggunaan dan resep, Anda harus memiliki minimal 1.000 interaksi item untuk 25 pengguna unik dengan setidaknya dua interaksi masing-masing. Untuk rekomendasi kualitas, kami sarankan Anda memiliki minimal 50.000 interaksi item dari setidaknya 1.000 pengguna dengan dua atau lebih interaksi item masing-masing.

Jika Anda tidak yakin apakah Anda memiliki cukup data, Anda dapat mengimpor dan menganalisisnya dengan konsol Amazon Personalize. Untuk informasi selengkapnya, lihat Menganalisis kualitas dan kuantitas data di Amazon Personalisasi kumpulan data.

Apakah Anda memiliki arsitektur streaming acara real-time?

Jika Anda tidak memiliki cukup data interaksi item, Anda dapat menggunakan Amazon Personalize untuk mengumpulkan data peristiwa real-time tambahan. Dengan beberapa resep dan kasus

penggunaan, Amazon Personalize dapat belajar dari aktivitas terbaru pengguna Anda dan memperbarui rekomendasi saat mereka menggunakan aplikasi Anda.

Untuk informasi tentang merekam peristiwa, termasuk bagaimana peristiwa memengaruhi rekomendasi, daftar layanan pelacakan peristiwa pihak ketiga, dan implementasi sampel, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi.

Apakah data Anda dioptimalkan untuk Amazon Personalize?

Kami sarankan Anda memeriksa hal-hal berikut dalam data Anda:

- Periksa nilai yang hilang. Kami merekomendasikan bahwa minimal 70% dari catatan Anda memiliki data untuk setiap atribut. Kami merekomendasikan kolom yang memungkinkan nilai nol setidaknya 70% selesai.
- Memperbaiki ketidakakuratan atau masalah dalam data Anda, seperti konvensi penamaan yang tidak konsisten, kategori duplikat untuk item, IDs ketidakcocokan di seluruh kumpulan data, atau duplikat. IDs Masalah-masalah ini dapat berdampak negatif pada rekomendasi atau mengarah pada perilaku yang tidak terduga. Misalnya, Anda mungkin memiliki keduanya “N/A” and “Not Applicable” in your data, but filter out recommendations based on only “N/A”. Item bertanda “Tidak Berlaku” tidak akan dihapus oleh filter.
- Jika item, pengguna, atau tindakan dapat memiliki beberapa kategori, seperti film dengan beberapa genre, gabungkan nilai kategoris menjadi satu atribut dan pisahkan setiap nilai dengan operator |. Misalnya, data GENRES film mungkin Action | Adventure | Thriller.
- Hindari memiliki lebih dari 1000 kategori yang mungkin untuk kolom (kecuali kolom berisi data hanya untuk tujuan penyaringan).

Untuk daftar lengkap rekomendasi data, dan petunjuk tentang cara menggunakan Amazon Personalize untuk mengidentifikasi masalah, lihat. Menganalisis kualitas dan kuantitas data di Amazon Personalisasi kumpulan data

Apakah Anda mengumpulkan data opsional yang dapat meningkatkan rekomendasi?

Data berikut dapat membantu meningkatkan relevansi rekomendasi Anda.

- Jenis peristiwa (diperlukan untuk semua kasus penggunaan grup kumpulan data Domain)

- Nilai acara
- Metadata kontekstual
- Metadata item dan pengguna
- Data interaksi tindakan (hanya digunakan oleh resep PERSONALIZED_ACTIONS)

Untuk informasi selengkapnya tentang jenis data yang dapat digunakan Amazon Personalize, lihat. Jenis data yang dapat digunakan Amazon Personalize

Apakah Anda memiliki rencana untuk menguji rekomendasi Anda?

Anda dapat menggunakan A/B pengujian untuk membandingkan hasil berbagai kelompok pengguna yang berinteraksi dengan rekomendasi dari model yang berbeda. A/B pengujian dapat membantu Anda membandingkan strategi rekomendasi yang berbeda dan melihat apakah rekomendasi membantu Anda mencapai tujuan bisnis Anda. Untuk informasi selengkapnya, lihat Mengukur dampak rekomendasi dengan A/B pengujian.

Apakah Anda memiliki tujuan bisnis tambahan?

Dalam beberapa kasus, Anda mungkin memiliki tujuan selain menghasilkan rekomendasi yang relevan untuk pengguna Anda. Misalnya, Anda mungkin ingin memaksimalkan pendapatan, atau mempromosikan jenis item tertentu dari kategori tertentu. Fitur Amazon Personalize berikut dapat membantu:

- Promosi: Anda dapat menggunakan promosi untuk memastikan persentase tertentu dari item memenuhi kebutuhan bisnis Anda. Untuk informasi selengkapnya, lihat Mempromosikan item dalam rekomendasi waktu nyata.
- Mengoptimalkan tujuan bisnis: Untuk beberapa resep grup kumpulan data kustom, Anda dapat mengoptimalkan solusi untuk tujuan khusus, seperti memaksimalkan menit streaming atau meningkatkan pendapatan. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi untuk tujuan tambahan.
- Rekomendasi penyaringan. Gunakan filter untuk menerapkan aturan bisnis pada rekomendasi. Anda dapat menggunakan filter untuk menyertakan atau mengecualikan jenis item tertentu dari rekomendasi. Lihat informasi yang lebih lengkap di Memfilter rekomendasi dan segmen pengguna.

Membuat grup kumpulan data Amazon Personalize

Setelah Anda membuat file JSON skema untuk data Anda, Anda siap untuk membuat grup dataset. Di Amazon Personalize, grup kumpulan data adalah wadah untuk sumber daya Amazon Personalize, termasuk kumpulan data, pemberi rekomendasi domain, dan sumber daya khusus. Grup kumpulan data mengatur sumber daya Anda ke dalam koleksi independen, di mana sumber daya dari satu grup kumpulan data tidak dapat memengaruhi sumber daya dalam grup kumpulan data lainnya.

Anda membuat grup kumpulan data untuk setiap domain bisnis Anda. Misalnya, Anda mungkin memiliki aplikasi yang memberikan rekomendasi untuk streaming video dan aplikasi lain yang memberikan rekomendasi untuk buku audio. Di Amazon Personalize, Anda akan membuat grup kumpulan data untuk setiap aplikasi. Dengan cara ini, data dari satu aplikasi tidak memengaruhi rekomendasi yang dihasilkan Amazon Personalize untuk aplikasi lain.

Anda dapat membuat grup kumpulan data Domain atau grup kumpulan data kustom:

- Dengan grup kumpulan data Domain, Anda membuat sumber daya yang telah dikonfigurasi sebelumnya dan dioptimalkan untuk berbagai kasus penggunaan. Saat membuat grup kumpulan data, Anda menjadikannya grup kumpulan data Domain dengan menentukan domain VIDEO_ON_DEMAND atau ECOMMERCE.

  Jika Anda memiliki aplikasi streaming video atau e-commerce, kami sarankan Anda membuat grup kumpulan data Domain. Anda masih dapat menambahkan sumber daya khusus, seperti solusi dan versi solusi yang dilatih untuk kasus penggunaan khusus. Anda tidak dapat membuat sumber daya tindakan terbaik berikutnya, termasuk kumpulan data Tindakan dan Interaksi Tindakan, dalam grup kumpulan data domain.
- Grup kumpulan data khusus hanya menyertakan sumber daya kustom yang Anda konfigurasikan tergantung pada kasus penggunaan Anda. Dengan sumber daya khusus, Anda melatih dan menerapkan solusi dan versi solusi yang dapat dikonfigurasi (model rekomendasi Amazon Personalize yang terlatih) berdasarkan kebutuhan bisnis Anda. Jika tidak memiliki aplikasi VIDEO_ON_DEMAND atau ECOMMERCE, sebaiknya Anda membuat grup dataset kustom. Jika tidak, sebaiknya mulai dengan grup kumpulan data Domain dan menambahkan sumber daya khusus seperlunya.

Anda dapat membuat grup kumpulan data dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau. AWS SDKs

Topik

- Membuat grup dataset (konsol)
- Membuat grup dataset ()AWS CLI
- Membuat grup dataset ()AWS SDKs

Membuat grup dataset (konsol)

Buat grup kumpulan data dengan menentukan nama grup kumpulan data di konsol Amazon Personalize.

Untuk membuat grup dataset

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih Buat grup kumpulan data.
3. Jika ini adalah pertama kalinya Anda menggunakan Amazon Personalize, pada halaman Buat grup kumpulan data, di grup Dataset baru, pilih Memulai.
4. Di detail grup Dataset, untuk nama grup Dataset, tentukan nama untuk grup kumpulan data Anda.
5. Pilih Domain Anda:
   - Pilih E-commerce untuk membuat grup dataset Domain ECOMMERCE.
   - Pilih Video sesuai permintaan untuk membuat grup kumpulan data Domain VIDEO_ON_DEMAND.
   - Pilih Kustom untuk membuat grup kumpulan data khusus hanya dengan sumber daya khusus, seperti solusi, kampanye, dan pekerjaan inferensi batch.
6. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya
7. Pilih Buat grup kumpulan data. Halaman Ikhtisar ditampilkan. Anda sekarang siap untuk membuat skema dan dataset. Lihat Membuat skema dan dataset.

Membuat grup dataset ()AWS CLI

Untuk membuat grup dataset, gunakan `create-dataset-group` operasi. Untuk membuat grup dataset Domain, untuk menentukan domain `ECOMMERCE` atau`VIDEO_ON_DEMAND`. Untuk membuat grup kumpulan data Kustom, jangan tentukan domain. Anda dapat menggunakan parameter Tag untuk menandai sumber daya secara opsional di Amazon Personalize. Untuk sampel lihatMenambahkan tag (AWS CLI).

Kode berikut membuat grup dataset Domain untuk `VIDEO_ON_DEMAND` domain.

```
aws personalize create-dataset-group \
--name dataset-group-name \
--domain VIDEO_ON_DEMAND
```

Jika berhasil, grup kumpulan data Amazon Resource Name (ARN) ditampilkan sebagai berikut.

```
{
  "datasetGroupArn": "arn:aws:personalize:us-west-2:acct-id:dataset-group/
DatasetGroupName"
}
```

Catat nilai ini untuk penggunaan masa depan. Untuk menampilkan grup kumpulan data yang Anda buat, gunakan `describe-dataset-group` perintah dan tentukan ARN grup kumpulan data yang dikembalikan.

```
aws personalize describe-dataset-group \
--dataset-group-arn dataset group arn
```

Grup dataset dan propertinya ditampilkan sebagai berikut.

```
{
    "datasetGroup": {
        "name": "DatasetGroupName",
        "datasetGroupArn": "arn:aws:personalize:us-west-2:acct-id:dataset-group/
DatasetGroupName",
        "status": "ACTIVE",
        "creationDateTime": 1542392161.262,
        "lastUpdatedDateTime": 1542396513.377
```

```
    }
}
```

Saat grup kumpulan data AKTIF, Anda siap membuat skema dan kumpulan data. `status` Lihat Membuat skema dan dataset.

Membuat grup dataset ()AWS SDKs

Gunakan kode berikut untuk membuat grup dataset Domain. Beri nama grup dataset Domain, dan untuk`domain`, tentukan salah satu `ECOMMERCE` atau`VIDEO_ON_DEMAND`. Untuk membuat grup dataset kustom, ubah kode untuk menghapus parameter domain.

Untuk informasi selengkapnya tentang operasi API, lihat CreateDatasetGroup di bagian referensi API. Anda dapat menggunakan parameter Tag untuk menandai sumber daya secara opsional di Amazon Personalize. Untuk sampel lihatMenambahkan tag (AWS SDKs).

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.create_dataset_group(
  name = 'dataset group name',
  domain = 'business domain'
)
dsg_arn = response['datasetGroupArn']

description = personalize.describe_dataset_group(datasetGroupArn = dsg_arn)
['datasetGroup']

print('Name: ' + description['name'])
print('ARN: ' + description['datasetGroupArn'])
print('Status: ' + description['status'])
```

SDK for Java 2.x

```
public static String createDomainDatasetGroup(PersonalizeClient personalizeClient,
                                               String datasetGroupName,
                                               String domain) {
```

```
    try {
        CreateDatasetGroupRequest createDatasetGroupRequest =
 CreateDatasetGroupRequest.builder()
                .name(datasetGroupName)
                .domain(domain)
                .build();
        return
 personalizeClient.createDatasetGroup(createDatasetGroupRequest).datasetGroupArn();
    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateDatasetGroupCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the domain dataset group parameters.
export const domainDatasetGroupParams = {
  name: "NAME" /* required */,
  domain:
    "DOMAIN" /* required for a domain dsg, specify ECOMMERCE or VIDEO_ON_DEMAND */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetGroupCommand(domainDatasetGroupParams),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

DescribeDatasetGroupOperasi mengembalikan `datasetGroupArn` dan status operasi. Saat grup kumpulan data AKTIF, Anda siap membuat skema dan kumpulan data. `status` Lihat Membuat skema dan dataset.

Membuat skema dan dataset

Setelah membuat grup kumpulan data, Anda siap membuat skema Amazon Personalize dan kumpulan data untuk setiap jenis data yang Anda impor. Skema memberi tahu Amazon Personalize tentang struktur data Anda dan memungkinkan Amazon Personalize untuk mengurai data. Saat membuat skema di Amazon Personalize, Anda menggunakan file JSON yang Anda buat. Membuat file JSON skema untuk skema Amazon Personalize

Dataset adalah wadah untuk melatih data di Amazon Personalize. Jenis dataset yang berbeda memiliki persyaratan yang berbeda. Anda membuat kumpulan data untuk setiap jenis data yang Anda impor. Untuk informasi tentang berbagai jenis kumpulan data dan cara menyiapkan data Anda, lihat. Mempersiapkan data pelatihan untuk Amazon Personalize

Anda dapat membuat skema dan kumpulan data dengan konsol Amazon Personalize, (), AWS Command Line Interface atau.AWS CLI AWS SDKs Anda tidak dapat membuat sumber daya tindakan terbaik berikutnya, termasuk kumpulan data Tindakan dan Interaksi Tindakan, dalam grup kumpulan data domain.

> **Important**
>
> Setelah Anda membuat skema, Anda tidak dapat membuat perubahan pada skema. Namun, jika Anda menambahkan kolom baru, Anda dapat mengganti skema kumpulan data dengan yang baru. Untuk informasi selengkapnya, lihat Mengganti skema dataset untuk menambahkan kolom baru.

Topik

- Membuat kumpulan data dan skema (konsol)
- Membuat dataset dan skema ()AWS CLI
- Membuat dataset dan skema ()AWS SDKs

Membuat kumpulan data dan skema (konsol)

Jika ini adalah kumpulan data pertama Anda di grup kumpulan data Anda, jenis kumpulan data pertama Anda akan menjadi kumpulan data interaksi Item. Untuk membuat kumpulan data interaksi Item di konsol, tentukan nama kumpulan data lalu tentukan skema JSON dalam format Avro. Jika

ini bukan kumpulan data pertama Anda dalam grup kumpulan data ini, pilih jenis kumpulan data lalu tentukan nama dan skema.

Untuk informasi tentang persyaratan Amazon Personalisasi kumpulan data, lihat. Mempersiapkan data pelatihan untuk Amazon Personalize Jika Anda baru saja selesai Membuat grup kumpulan data Amazon Personalize dan Anda sudah membuat dataset Anda, lewati ke langkah 4 dalam prosedur ini.

Untuk membuat dataset dan skema

1. Jika Anda belum melakukannya, ikuti petunjuk Membuat file JSON skema untuk skema Amazon Personalize untuk membuat file JSON skema yang menguraikan data Anda.
2. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
3. Pada halaman grup Dataset, pilih grup kumpulan data yang Anda buat. Membuat grup kumpulan data Amazon Personalize
4. Pada Langkah 1. Buat kumpulan data dan impor data pilih Buat kumpulan data dan pilih jenis kumpulan data yang akan dibuat.
5. Pilih Impor data langsung ke Amazon Personalize dataset dan pilih Berikutnya.
6. Di detail Dataset, untuk nama Dataset, tentukan nama untuk kumpulan data Anda.
7. Untuk skema Dataset, pilih Buat skema baru atau Gunakan skema yang ada.
8. Jika Anda menggunakan skema yang ada, pilih skema yang ada untuk digunakan. Jika Anda membuat skema baru, beri nama skema dan tempel di skema JSON yang cocok dengan data Anda. Anda membuat file ini diMembuat file JSON skema untuk skema Amazon Personalize.
9. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya
10. Pilih Berikutnya dan ikuti petunjuk Mengimpor data pelatihan ke Amazon Personalize dataset untuk mengimpor data Anda.

Membuat dataset dan skema ()AWS CLI

Untuk membuat kumpulan data dan skema menggunakan AWS CLI, Anda menggunakan `create-schema` perintah (yang menggunakan operasi CreateSchema API) dan kemudian `create-dataset` (yang menggunakan operasi CreateDataset API).

Untuk membuat skema dan dataset

1. Jika Anda belum melakukannya, ikuti petunjuk Membuat file JSON skema untuk skema Amazon Personalize untuk membuat file JSON skema yang menguraikan data Anda.
2. Buat skema di Amazon Personalize dengan menjalankan perintah berikut. Setelah Anda membuat skema, Anda tidak dapat membuat perubahan pada skema. Ganti `schemaName` dengan nama skema, dan ganti `file://SchemaName.json` dengan lokasi file JSON Anda. Contoh menunjukkan file sebagai milik folder saat ini. Jika Anda membuat skema untuk kumpulan data dalam grup dataset Domain, tambahkan `domain` parameter dan atur ke atau. `ECOMMERCE` `VIDEO_ON_DEMAND` Untuk informasi selengkapnya tentang API, lihat CreateSchema.

   ```
   aws personalize create-schema \
     --name SchemaName \
     --schema file://SchemaName.json
   ```

   Skema Amazon Resource Name (ARN) ditampilkan, seperti yang ditunjukkan pada contoh berikut:

   ```
   {
     "schemaArn": "arn:aws:personalize:us-west-2:acct-id:schema/SchemaName"
   }
   ```

3. Buat dataset kosong dengan menjalankan perintah berikut. Berikan grup kumpulan data Amazon Resource Name (ARN) Membuat grup dataset ()AWS CLI dari dan skema ARN dari langkah sebelumnya. Nilai tipe dataset dapat berupa`Interactions`,,`Users`, `ItemsActions`, atau`Action_Interactions`. Untuk informasi selengkapnya tentang API, lihat CreateDataset.

   ```
   aws personalize create-dataset \
     --name Dataset Name \
     --dataset-group-arn Dataset Group ARN \
     --dataset-type Dataset Type \
     --schema-arn Schema Arn
   ```

   Dataset ARN ditampilkan, seperti yang ditunjukkan pada contoh berikut.

   ```
   {
     "datasetArn": "arn:aws:personalize:us-west-2:acct-id:dataset/DatasetName/
   INTERACTIONS"
   ```

```
}
```

4. Rekam kumpulan data ARN untuk digunakan nanti. Setelah Anda membuat dataset, Anda siap untuk mengimpor data pelatihan Anda. Lihat Mengimpor data pelatihan ke Amazon Personalize dataset.

Membuat dataset dan skema ()AWS SDKs

Untuk membuat kumpulan data dan skema menggunakan skema AWS SDKs, pertama-tama Anda menentukan skema dalam format Avro dan menambahkannya ke Amazon Personalize menggunakan operasi. CreateSchema Setelah Anda membuat skema, Anda tidak dapat membuat perubahan pada skema. Kemudian buat dataset menggunakan CreateDataset operasi.

Untuk membuat skema dan dataset

1. Jika Anda belum melakukannya, ikuti petunjuk Membuat file JSON skema untuk skema Amazon Personalize untuk membuat file JSON skema yang menguraikan data Anda.
2. Buat skema di Amazon Personalize dengan kode berikut. Tentukan nama untuk skema Anda dan jalur file untuk file JSON skema Anda. Jika Anda membuat skema untuk kumpulan data dalam grup dataset Domain, tambahkan `domain` parameter dan atur ke atau. `ECOMMERCE` `VIDEO_ON_DEMAND` Untuk informasi selengkapnya tentang API, lihat CreateSchema.

   SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

with open('schemaFile.json') as f:
    createSchemaResponse = personalize.create_schema(
        name = 'schema name',
        schema = f.read()
    )

schema_arn = createSchemaResponse['schemaArn']

print('Schema ARN:' + schema_arn )
```

SDK for Java 2.x

```
public static String createSchema(PersonalizeClient personalizeClient, String
 schemaName, String filePath) {

    String schema = null;

    try {
        schema = new String(Files.readAllBytes(Paths.get(filePath)));
    } catch (IOException e) {
        System.out.println(e.getMessage());
    }

    try {
        CreateSchemaRequest createSchemaRequest = CreateSchemaRequest.builder()
                .name(schemaName)
                .schema(schema)
                .build();

        String schemaArn =
 personalizeClient.createSchema(createSchemaRequest).schemaArn();
        System.out.println("Schema arn: " + schemaArn);

        return schemaArn;

    } catch(PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateSchemaCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

import fs from "node:fs";
```

```
const schemaFilePath = "SCHEMA_PATH";
let mySchema = "";

try {
  mySchema = fs.readFileSync(schemaFilePath).toString();
} catch (err) {
  mySchema = "TEST"; // For unit tests.
}
// Set the schema parameters.
export const createSchemaParam = {
  name: "NAME" /* required */,
  schema: mySchema /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSchemaCommand(createSchemaParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Amazon Personalize mengembalikan ARN skema baru. Rekam karena Anda akan membutuhkannya di langkah berikutnya.

3. Buat kumpulan data menggunakan CreateDataset operasi. Kode berikut menunjukkan cara membuat dataset. Tentukan Nama Sumber Daya Amazon (ARN) grup kumpulan data Anda, skema ARN dari langkah sebelumnya, dan tentukan jenis kumpulan data. Nilai tipe dataset dapat berupa`Interactions`,,`Users`, `ItemsActions`, atau`Action_Interactions`. Untuk informasi tentang berbagai jenis kumpulan data, lihat. Mempersiapkan data pelatihan untuk Amazon Personalize

   SDK for Python (Boto3)

```
import boto3
```

```
personalize = boto3.client('personalize')

response = personalize.create_dataset(
    name = 'dataset_name',
    schemaArn = 'schema_arn',
    datasetGroupArn = 'dataset_group_arn',
    datasetType = 'dataset_type'
)

print ('Dataset Arn: ' + response['datasetArn'])
```

SDK for Java 2.x

```
public static String createDataset(PersonalizeClient personalizeClient,
                                   String datasetName,
                                   String datasetGroupArn,
                                   String datasetType,
                                   String schemaArn) {
    try {
        CreateDatasetRequest request = CreateDatasetRequest.builder()
                .name(datasetName)
                .datasetGroupArn(datasetGroupArn)
                .datasetType(datasetType)
                .schemaArn(schemaArn).build();

        String datasetArn =
 personalizeClient.createDataset(request).datasetArn();
        System.out.println("Dataset " + datasetName + " created. Dataset ARN: "
 + datasetArn);

        return datasetArn;

    } catch(PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateDatasetCommand } from "@aws-sdk/client-personalize";
```

```
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the dataset's parameters.
export const createDatasetParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  datasetType: "DATASET_TYPE" /* required */,
  name: "NAME" /* required */,
  schemaArn: "SCHEMA_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetCommand(createDatasetParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Setelah Anda membuat dataset, Anda siap untuk mengimpor data pelatihan Anda. Lihat Mengimpor data pelatihan ke Amazon Personalize dataset.

Mengimpor data pelatihan ke Amazon Personalize dataset

Setelah Anda selesai membuat skema dan dataset, Anda siap untuk mengimpor data pelatihan Anda ke dalam dataset. Saat mengimpor data, Anda dapat memilih untuk mengimpor catatan secara massal, satu per satu, atau keduanya.

- Impor massal melibatkan impor sejumlah besar catatan sejarah sekaligus. Anda dapat menyiapkan data massal sendiri, dan mengimpornya langsung ke Amazon Personalisasi dari file CSV di Amazon S3. Untuk informasi tentang cara menyiapkan data Anda, lihatMempersiapkan data pelatihan untuk Amazon Personalize. Jika Anda memerlukan bantuan untuk mempersiapkan data, Anda dapat menggunakan SageMaker AI Data Wrangler untuk menyiapkan dan mengimpor interaksi item massal, pengguna, dan data item Anda. Untuk informasi selengkapnya, lihat Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler.
- Jika tidak memiliki data massal, Anda dapat menggunakan operasi impor individual untuk mengumpulkan data dan streaming peristiwa hingga memenuhi persyaratan pelatihan Amazon Personalize dan persyaratan data kasus atau resep penggunaan domain Anda. Untuk informasi tentang merekam acara, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi. Untuk informasi tentang mengimpor catatan individual, lihatMengimpor catatan individual ke dalam kumpulan data Amazon Personalize.

Setelah mengimpor data ke dalam kumpulan data Amazon Personalize, Anda dapat menganalisisnya, mengekspornya ke bucket Amazon S3, memperbaruinya, atau menghapusnya dengan menghapus kumpulan data.

Jika Anda mengimpor item, pengguna, atau tindakan dengan ID yang sama dengan catatan yang sudah ada di kumpulan data Anda, Amazon Personalize menggantinya dengan catatan baru. Jika Anda merekam dua interaksi item atau peristiwa interaksi tindakan dengan stempel waktu yang sama persis dan properti yang identik, Amazon Personalize hanya menyimpan satu peristiwa.

Seiring pertumbuhan katalog Anda, perbarui data historis Anda dengan operasi impor massal atau data individual tambahan. Untuk rekomendasi waktu nyata, perbarui kumpulan data interaksi Item Anda dengan perilaku pengguna Anda. Anda melakukan ini dengan merekam peristiwa interaksi real-time dengan pelacak peristiwa dan PutEvents operasi. Untuk informasi selengkapnya, lihat Merekam peristiwa waktu nyata untuk memengaruhi rekomendasi

Setelah mengimpor data, Anda siap membuat pemberi rekomendasi domain (untuk grup kumpulan data Domain) atau sumber daya khusus (untuk grup kumpulan data kustom) untuk melatih model pada data Anda. Anda menggunakan sumber daya ini untuk menghasilkan rekomendasi. Untuk informasi selengkapnya, lihat Rekomendasi domain di Amazon Personalisasi atau Sumber daya khusus untuk melatih dan menerapkan model Amazon Personalize.

Topik

- Mengimpor data massal ke Amazon Personalisasi dengan pekerjaan impor kumpulan data
- Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler
- Mengimpor catatan individual ke dalam kumpulan data Amazon Personalize

Mengimpor data massal ke Amazon Personalisasi dengan pekerjaan impor kumpulan data

Setelah Anda memformat data input Anda (lihatMempersiapkan data pelatihan untuk Amazon Personalize) dan selesaiMembuat skema dan dataset, Anda siap untuk mengimpor data massal Anda dengan pekerjaan impor dataset. Pekerjaan impor dataset adalah alat impor massal yang mengisi kumpulan data dengan data dari Amazon S3.

Untuk mengimpor data dari Amazon S3, file CSV Anda harus berada di bucket Amazon S3 dan Anda harus memberikan izin Amazon Personalize untuk mengakses sumber daya Amazon S3 Anda:

- Untuk informasi tentang mengunggah file ke Amazon S3, lihat Mengunggah File dan Folder dengan Menggunakan Seret dan Jatuhkan di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.
- Untuk informasi tentang memberikan Amazon Personalisasi akses ke file Anda di Amazon S3, lihat. Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3

  Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.

Anda dapat membuat tugas impor kumpulan data menggunakan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau. AWS SDKs Jika sebelumnya Anda membuat pekerjaan impor kumpulan data untuk kumpulan data, Anda dapat menggunakan pekerjaan impor kumpulan

data baru untuk menambah atau mengganti data massal yang ada. Untuk informasi selengkapnya, lihat Memperbarui data dalam kumpulan data setelah pelatihan.

Jika Anda mengimpor item, pengguna, atau tindakan dengan ID yang sama dengan catatan yang sudah ada di kumpulan data Anda, Amazon Personalize menggantinya dengan catatan baru. Jika Anda merekam dua interaksi item atau peristiwa interaksi tindakan dengan stempel waktu yang sama persis dan properti yang identik, Amazon Personalize hanya menyimpan satu peristiwa.

Setelah mengimpor data, Anda siap membuat pemberi rekomendasi domain (untuk grup kumpulan data Domain) atau sumber daya khusus (untuk grup kumpulan data kustom) untuk melatih model pada data Anda. Anda menggunakan sumber daya ini untuk menghasilkan rekomendasi. Untuk informasi selengkapnya, lihat Rekomendasi domain di Amazon Personalisasi atau Sumber daya khusus untuk melatih dan menerapkan model Amazon Personalize.

Topik

- Mode impor
- Membuat pekerjaan impor dataset (konsol)
- Membuat pekerjaan impor dataset ()AWS CLI
- Membuat pekerjaan impor dataset ()AWS SDKs

Mode impor

Jika Anda telah membuat pekerjaan impor untuk kumpulan data, Anda dapat mengonfigurasi cara Amazon Personalize menambahkan catatan baru Anda. Untuk melakukan ini, Anda menentukan mode impor untuk pekerjaan impor dataset Anda. Jika Anda belum mengimpor data massal, bidang mode Impor tidak tersedia di konsol dan Anda hanya dapat menentukan `FULL` dalam operasi `CreateDatasetImportJob` API. Defaultnya adalah pengganti penuh.

- Untuk menimpa semua data massal yang ada di kumpulan data Anda, pilih Ganti data yang ada di konsol Amazon Personalize atau `FULL` tentukan dalam CreateDatasetImportJob operasi API. Ini tidak menggantikan data yang Anda impor satu per satu, termasuk peristiwa yang direkam secara real time.
- Untuk menambahkan catatan ke data yang ada di kumpulan data Anda, pilih Tambahkan ke data yang ada atau tentukan `INCREMENTAL` dalam operasi `CreateDatasetImportJob` API. Amazon Personalize menggantikan catatan apa pun dengan ID yang sama dengan yang baru.

> **Note**
>
> Untuk menambahkan data ke kumpulan data interaksi Item atau Kumpulan data interaksi tindakan dengan pekerjaan impor kumpulan data, Anda harus memiliki minimal 1000 interaksi item baru atau catatan interaksi tindakan.

Membuat pekerjaan impor dataset (konsol)

> **Important**
>
> Secara default, pekerjaan impor dataset menggantikan data yang ada dalam kumpulan data yang Anda impor secara massal. Jika Anda sudah mengimpor data massal, Anda dapat menambahkan data dengan mengubah mode impor pekerjaan.

Untuk mengimpor data massal ke dalam kumpulan data dengan konsol Amazon Personalize, buat pekerjaan impor kumpulan data dengan nama, peran layanan IAM, dan lokasi data Anda.

Jika Anda baru saja membuat kumpulan data diMembuat skema dan dataset, lewati ke langkah 5.

Untuk mengimpor catatan massal (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Anda. Ikhtisar grup kumpulan data ditampilkan.
3. Di panel navigasi, pilih Kumpulan data dan pilih kumpulan data yang ingin Anda impor data massal.
4. Di pekerjaan impor Dataset, pilih Buat pekerjaan impor set data.
5. Jika ini adalah pekerjaan impor kumpulan data pertama Anda, untuk Sumber impor data pilih Impor data dari S3.
6. Untuk nama pekerjaan impor Dataset, tentukan nama untuk pekerjaan impor Anda.
7. Jika Anda sudah mengimpor data massal, untuk mode Impor, pilih cara memperbarui kumpulan data. Pilih salah satu Ganti data yang ada atau Tambahkan ke data yang ada. data. Opsi ini

tidak muncul jika ini adalah pekerjaan pertama Anda untuk kumpulan data. Untuk informasi selengkapnya, lihat Memperbarui data dalam kumpulan data setelah pelatihan.

8. Di Sumber impor data, untuk Lokasi Data, tentukan tempat file data Anda disimpan di Amazon S3. Gunakan sintaks berikut:

   ```
   s3:/amzn-s3-demo-bucket/<folder path>/<CSV filename>
   ```

   Jika file CSV Anda berada di folder di bucket Amazon S3 dan Anda ingin mengunggah beberapa file CSV ke kumpulan data dengan satu tugas impor kumpulan data, Anda dapat menentukan jalur ke folder tersebut. Amazon Personalize hanya menggunakan file di tingkat pertama folder Anda, tidak menggunakan data apa pun di sub folder apa pun. Gunakan sintaks berikut dengan nama folder / setelah:

   ```
   s3:/amzn-s3-demo-bucket/<folder path>/
   ```

9. Dalam peran IAM, pilih untuk membuat peran baru atau menggunakan yang sudah ada. Jika Anda menyelesaikan prasyarat, pilih Gunakan peran layanan yang ada dan tentukan peran yang Anda buat. Membuat peran IAM untuk Amazon Personalize

10. Jika Anda membuat atribusi metrik dan ingin memublikasikan metrik yang terkait dengan pekerjaan ini ke Amazon S3, di Publikasikan metrik peristiwa ke S3 pilih Publikasikan metrik untuk pekerjaan impor ini.

    Jika Anda belum membuatnya dan ingin mempublikasikan metrik untuk pekerjaan ini, pilih Buat atribusi metrik untuk membuat yang baru di tab lain. Setelah membuat atribusi metrik, Anda dapat kembali ke layar ini dan menyelesaikan pembuatan pekerjaan impor.

    Untuk informasi lebih lanjut tentang atribusi metrik, lihatMengukur dampak rekomendasi Amazon Personalize.

11. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya

12. Pilih Mulai impor. Pekerjaan impor data dimulai dan halaman Ikhtisar Dasbor ditampilkan. Impor dataset selesai ketika status ditampilkan sebagai ACTIVE. Setelah mengimpor data ke dalam kumpulan data Amazon Personalize, Anda dapat menganalisisnya, mengekspornya ke bucket Amazon S3, memperbaruinya, atau menghapusnya dengan menghapus kumpulan data.

    Setelah mengimpor data, Anda siap membuat pemberi rekomendasi domain (untuk grup kumpulan data Domain) atau sumber daya khusus (untuk grup kumpulan data kustom) untuk

melatih model pada data Anda. Anda menggunakan sumber daya ini untuk menghasilkan rekomendasi. Untuk informasi selengkapnya, lihat Rekomendasi domain di Amazon Personalisasi atau Sumber daya khusus untuk melatih dan menerapkan model Amazon Personalize.

Membuat pekerjaan impor dataset ()AWS CLI

> **Important**
>
> Secara default, pekerjaan impor dataset menggantikan data yang ada dalam kumpulan data yang Anda impor secara massal. Jika Anda sudah mengimpor data massal, Anda dapat menambahkan data dengan mengubah mode impor pekerjaan.

Untuk mengimpor catatan massal menggunakan AWS CLI, buat pekerjaan impor dataset menggunakan CreateDatasetImportJob perintah. Jika sebelumnya Anda telah membuat pekerjaan impor kumpulan data untuk kumpulan data, Anda dapat menggunakan parameter mode impor untuk menentukan cara menambahkan data baru. Untuk informasi selengkapnya tentang memperbarui data massal yang ada, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Impor catatan massal (AWS CLI)

1. Buat pekerjaan impor dataset dengan menjalankan perintah berikut. Berikan Nama Sumber Daya Amazon (ARN) untuk kumpulan data Anda dan tentukan jalur ke bucket Amazon S3 tempat Anda menyimpan data pelatihan. Gunakan sintaks berikut untuk jalur:

   `s3://amzn-s3-demo-bucket/<folder path>/<CSV filename>`

   Jika file CSV Anda berada di folder di bucket Amazon S3 dan Anda ingin mengunggah beberapa file CSV ke kumpulan data dengan satu tugas impor kumpulan data, Anda dapat menentukan jalur ke folder tersebut. Amazon Personalize hanya menggunakan file di tingkat pertama folder Anda, tidak menggunakan data apa pun di sub folder apa pun. Gunakan sintaks berikut dengan nama folder / setelah:

   `s3://amzn-s3-demo-bucket/<folder path>/`

   Berikan peran AWS Identity and Access Management (IAM) Nama Sumber Daya Amazon (ARN) yang Anda buat. Membuat peran IAM untuk Amazon Personalize `import-`

modeDefaultnya adalahFULL. Untuk mengetahui informasi selengkapnya, lihat Memperbarui data dalam kumpulan data setelah pelatihan. Untuk informasi lebih lanjut tentang operasi, lihatCreateDatasetImportJob.

```
aws personalize create-dataset-import-job \
--job-name dataset import job name \
--dataset-arn dataset arn \
--data-source dataLocation=s3://amzn-s3-demo-bucket/filename \
--role-arn roleArn \
--import-mode FULL
```

Pekerjaan impor dataset ARN ditampilkan, seperti yang ditunjukkan pada contoh berikut.

```
{
  "datasetImportJobArn": "arn:aws:personalize:us-west-2:acct-id:dataset-import-job/
DatasetImportJobName"
}
```

2. Periksa status dengan menggunakan `describe-dataset-import-job` perintah. Berikan ARN pekerjaan impor dataset yang dikembalikan pada langkah sebelumnya. Untuk informasi lebih lanjut tentang operasi, lihatDescribeDatasetImportJob.

```
aws personalize describe-dataset-import-job \
--dataset-import-job-arn dataset import job arn
```

Properti pekerjaan impor dataset, termasuk statusnya, ditampilkan. Awalnya, `status` ditampilkan sebagai CREATE PENDING.

```
{
  "datasetImportJob": {
      "jobName": "Dataset Import job name",
      "datasetImportJobArn": "arn:aws:personalize:us-west-2:acct-id:dataset-import-
job/DatasetImportJobArn",
      "datasetArn": "arn:aws:personalize:us-west-2:acct-id:dataset/
DatasetGroupName/INTERACTIONS",
      "dataSource": {
          "dataLocation": "s3://amzn-s3-demo-bucket/ratings.csv"
      },
      "importMode": "FULL",
      "roleArn": "role-arn",
      "status": "CREATE PENDING",
```

```
        "creationDateTime": 1542392161.837,
        "lastUpdatedDateTime": 1542393013.377
    }
}
```

Impor dataset selesai ketika status ditampilkan sebagai ACTIVE. Setelah mengimpor data ke dalam kumpulan data Amazon Personalize, Anda dapat menganalisisnya, mengekspornya ke bucket Amazon S3, memperbaruinya, atau menghapusnya dengan menghapus kumpulan data.

Setelah mengimpor data, Anda siap membuat pemberi rekomendasi domain (untuk grup kumpulan data Domain) atau sumber daya khusus (untuk grup kumpulan data kustom) untuk melatih model pada data Anda. Anda menggunakan sumber daya ini untuk menghasilkan rekomendasi. Untuk informasi selengkapnya, lihat Rekomendasi domain di Amazon Personalisasi atau Sumber daya khusus untuk melatih dan menerapkan model Amazon Personalize.

Membuat pekerjaan impor dataset ()AWS SDKs

> **Important**
>
> Secara default, pekerjaan impor dataset menggantikan data yang ada dalam kumpulan data yang Anda impor secara massal. Jika Anda sudah mengimpor data massal, Anda dapat menambahkan data dengan mengubah mode impor pekerjaan.

Untuk mengimpor data, buat pekerjaan impor dataset dengan CreateDatasetImportJob operasi. Kode berikut menunjukkan cara membuat pekerjaan impor dataset.

Berikan nama pekerjaan, setel Nama Sumber Daya Amazon (ARN) dari kumpulan data Anda, dan setel ke path `dataLocation` ke bucket Amazon S3 tempat Anda menyimpan data pelatihan. `datasetArn` Gunakan sintaks berikut untuk jalur:

`s3:/amzn-s3-demo-bucket/<folder path>/<CSV filename>.csv`

Jika file CSV Anda berada di folder di bucket Amazon S3 dan Anda ingin mengunggah beberapa file CSV ke kumpulan data dengan satu tugas impor kumpulan data, Anda dapat menentukan jalur ke folder tersebut. Amazon Personalize hanya menggunakan file di tingkat pertama folder Anda, tidak menggunakan data apa pun di sub folder apa pun. Gunakan sintaks berikut dengan nama folder `/` setelah:

`s3:/amzn-s3-demo-bucket/<folder path>/`

Untuk ituroleArn, tentukan peran AWS Identity and Access Management (IAM) yang memberikan izin Amazon Personalize untuk mengakses bucket S3 Anda. Lihat Membuat peran IAM untuk Amazon Personalize. importModeDefaultnya adalahFULL. Ini menggantikan semua data massal dalam kumpulan data. Untuk menambahkan data, atur keINCREMENTAL. Untuk informasi selengkapnya tentang memperbarui data massal yang ada, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.create_dataset_import_job(
    jobName = 'YourImportJob',
    datasetArn = 'dataset_arn',
    dataSource = {'dataLocation':'s3://amzn-s3-demo-bucket/filename.csv'},
    roleArn = 'role_arn',
    importMode = 'FULL'
)

dsij_arn = response['datasetImportJobArn']

print ('Dataset Import Job arn: ' + dsij_arn)

description = personalize.describe_dataset_import_job(
    datasetImportJobArn = dsij_arn)['datasetImportJob']

print('Name: ' + description['jobName'])
print('ARN: ' + description['datasetImportJobArn'])
print('Status: ' + description['status'])
```

SDK for Java 2.x

```
public static String createPersonalizeDatasetImportJob(PersonalizeClient
 personalizeClient,
                                                        String jobName,
                                                        String datasetArn,
                                                        String s3BucketPath,
                                                        String roleArn,
```

```
                                                    ImportMode importMode) {

 long waitInMilliseconds = 60 * 1000;
 String status;
 String datasetImportJobArn;

 try {
     DataSource importDataSource = DataSource.builder()
             .dataLocation(s3BucketPath)
             .build();

     CreateDatasetImportJobRequest createDatasetImportJobRequest =
CreateDatasetImportJobRequest.builder()
             .datasetArn(datasetArn)
             .dataSource(importDataSource)
             .jobName(jobName)
             .roleArn(roleArn)
             .importMode(importMode)
             .build();

     datasetImportJobArn =
personalizeClient.createDatasetImportJob(createDatasetImportJobRequest)
             .datasetImportJobArn();

     DescribeDatasetImportJobRequest describeDatasetImportJobRequest =
DescribeDatasetImportJobRequest.builder()
             .datasetImportJobArn(datasetImportJobArn)
             .build();

     long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

     while (Instant.now().getEpochSecond() < maxTime) {

         DatasetImportJob datasetImportJob = personalizeClient
                 .describeDatasetImportJob(describeDatasetImportJobRequest)
                 .datasetImportJob();

         status = datasetImportJob.status();
         System.out.println("Dataset import job status: " + status);

         if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
             break;
         }
         try {
```

```
                Thread.sleep(waitInMilliseconds);
            } catch (InterruptedException e) {
                System.out.println(e.getMessage());
            }
        }
        return datasetImportJobArn;

    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { CreateDatasetImportJobCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

// Set the dataset import job parameters.
export const datasetImportJobParam = {
  datasetArn: 'DATASET_ARN', /* required */
  dataSource: {
    dataLocation: 's3://amzn-s3-demo-bucket/<folderName>/<CSVfilename>.csv'  /*
 required */
  },
  jobName: 'NAME',           /* required */
  roleArn: 'ROLE_ARN',       /* required */
  importMode: "FULL"         /* optional, default is FULL */
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(new
 CreateDatasetImportJobCommand(datasetImportJobParam));
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
```

```
    console.log("Error", err);
  }
};
run();
```

Tanggapan dari DescribeDatasetImportJob operasi mencakup status operasi.

Anda harus menunggu hingga status berubah menjadi AKTIF sebelum Anda dapat menggunakan data untuk melatih model.

Impor dataset selesai ketika status ditampilkan sebagai ACTIVE. Setelah mengimpor data ke dalam kumpulan data Amazon Personalize, Anda dapat menganalisisnya, mengekspornya ke bucket Amazon S3, memperbaruinya, atau menghapusnya dengan menghapus kumpulan data.

Setelah mengimpor data, Anda siap membuat pemberi rekomendasi domain (untuk grup kumpulan data Domain) atau sumber daya khusus (untuk grup kumpulan data kustom) untuk melatih model pada data Anda. Anda menggunakan sumber daya ini untuk menghasilkan rekomendasi. Untuk informasi selengkapnya, lihat Rekomendasi domain di Amazon Personalisasi atau Sumber daya khusus untuk melatih dan menerapkan model Amazon Personalize.

Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler

> **Important**
>
> Saat Anda menggunakan Data Wrangler, Anda dikenakan biaya AI SageMaker . Untuk daftar lengkap biaya dan harga, lihat tab Data Wrangler dari harga Amazon SageMaker AI. Untuk menghindari biaya tambahan, setelah selesai, matikan instans Data Wrangler Anda. Untuk informasi selengkapnya, lihat Matikan Data Wrangler.

Setelah membuat grup kumpulan data, Anda dapat menggunakan Amazon SageMaker AI Data Wrangler (Data Wrangler) untuk mengimpor data dari 40+ sumber ke kumpulan data Amazon Personalize. Data Wrangler adalah fitur Amazon SageMaker AI Studio Classic yang menyediakan end-to-end solusi untuk mengimpor, menyiapkan, mengubah, dan menganalisis data. Anda tidak dapat menggunakan Data Wrangler untuk menyiapkan dan mengimpor data ke dalam kumpulan data Tindakan atau kumpulan data interaksi Tindakan.

Saat Anda menggunakan Data Wrangler untuk menyiapkan dan mengimpor data, Anda menggunakan aliran data. Aliran data mendefinisikan serangkaian langkah persiapan data pembelajaran mesin, dimulai dengan mengimpor data. Setiap kali Anda menambahkan langkah ke alur Anda, Data Wrangler mengambil tindakan pada data Anda, seperti mengubahnya atau menghasilkan visualisasi.

Berikut ini adalah beberapa langkah yang dapat Anda tambahkan ke alur Anda untuk menyiapkan data untuk Amazon Personalize:

- Wawasan: Anda dapat menambahkan Amazon Personalisasi langkah-langkah wawasan tertentu ke alur Anda. Wawasan ini dapat membantu Anda mempelajari tentang data Anda dan tindakan apa yang dapat Anda ambil untuk memperbaikinya.
- Visualisasi: Anda dapat menambahkan langkah-langkah visualisasi untuk menghasilkan grafik seperti histogram dan plot pencar. Grafik dapat membantu Anda menemukan masalah dalam data Anda, seperti outlier atau nilai yang hilang.
- Transformasi: Anda dapat menggunakan Amazon Personalize langkah-langkah transformasi spesifik dan umum untuk memastikan data Anda memenuhi persyaratan Amazon Personalize. Transformasi Amazon Personalize membantu Anda memetakan kolom data ke kolom yang diperlukan tergantung pada jenis dataset Amazon Personalize.

Jika Anda harus meninggalkan Data Wrangler sebelum mengimpor data ke Amazon Personalize, Anda dapat kembali ke tempat yang Anda tinggalkan dengan memilih jenis kumpulan data yang sama saat meluncurkan Data Wrangler dari konsol Amazon Personalize. Atau Anda dapat mengakses Data Wrangler langsung melalui SageMaker AI Studio Classic.

Kami menyarankan Anda mengimpor data dari Data Wrangler ke Amazon Personalize sebagai berikut. Langkah transformasi, visualisasi, dan analisis bersifat opsional, dapat diulang, dan dapat diselesaikan dalam urutan apa pun.

1. Siapkan izin - Siapkan izin untuk Amazon Personalize SageMaker dan peran layanan AI. Dan siapkan izin untuk pengguna Anda.
2. Luncurkan Data Wrangler di SageMaker AI Studio Classic dari konsol Amazon Personalize - Gunakan konsol Amazon Personalize untuk mengonfigurasi SageMaker domain AI dan meluncurkan Data Wrangler di AI Studio Classic. SageMaker
3. Impor data Anda ke Data Wrangler - Impor data dari 40+ sumber ke Data Wrangler. Sumber termasuk AWS layanan, seperti Amazon Redshift, Amazon EMR, atau Amazon Athena, dan pihak ke-3 seperti Snowflake atau. DataBricks

4. Ubah data Anda - Gunakan Data Wrangler untuk mengubah data Anda agar memenuhi persyaratan Amazon Personalize.
5. Memvisualisasikan dan menganalisis data Anda - Gunakan Data Wrangler untuk memvisualisasikan data Anda dan menganalisisnya melalui Amazon Personalisasi wawasan tertentu.
6. Memproses dan mengimpor data ke Amazon Personalize - Gunakan notebook SageMaker AI Studio Classic Jupyter untuk mengimpor data yang diproses ke Amazon Personalize.

Informasi tambahan

Sumber daya berikut memberikan informasi tambahan tentang penggunaan Amazon SageMaker AI Data Wrangler dan Amazon Personalize.

- Untuk tutorial yang memandu Anda melalui pemrosesan dan transformasi kumpulan data sampel, lihat Demo: Panduan Data Wrangler Titanic Dataset di Panduan Pengembang Amazon AI. SageMaker Tutorial ini memperkenalkan bidang dan fungsi Data Wrangler.
- Untuk informasi tentang orientasi ke domain Amazon SageMaker AI, lihat Cepat onboard ke Amazon SageMaker AI Domain di Panduan Pengembang Amazon SageMaker AI.
- Untuk informasi tentang persyaratan data Amazon Personalisasi, lihat. Mempersiapkan data pelatihan untuk Amazon Personalize

Menyiapkan izin

Untuk menyiapkan data dengan Data Wrangler, Anda harus mengatur izin berikut:

- Buat peran layanan untuk Amazon Personalisasi: Jika Anda belum melakukannya, lengkapi petunjuk Menyiapkan Amazon Personalisasi untuk membuat peran layanan IAM untuk Amazon Personalisasi. Peran ini harus memiliki `GetObject` dan `ListBucket` izin untuk bucket Amazon S3 yang menyimpan data yang Anda proses. Dan itu harus memiliki izin untuk menggunakan AWS KMS kunci apa pun.

  Untuk informasi tentang pemberian Amazon Personalize akses ke bucket Amazon S3 Anda, lihat. Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3 Untuk informasi tentang pemberian Amazon Personalize akses ke kunci AWS KMS Anda, lihat. Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS

- Buat pengguna administratif dengan izin SageMaker AI: Administrator Anda harus memiliki akses penuh ke SageMaker AI dan harus dapat membuat domain SageMaker AI. Untuk informasi selengkapnya, lihat Membuat Pengguna dan Grup Administratif di Panduan Pengembang Amazon SageMaker AI.
- Buat peran eksekusi SageMaker AI: Buat peran eksekusi SageMaker AI dengan akses ke sumber daya SageMaker AI dan Amazon Personalisasi operasi impor data. Peran eksekusi SageMaker AI harus memiliki `AmazonSageMakerFullAccess`kebijakan yang terlampir. Jika Anda memerlukan izin Data Wrangler yang lebih terperinci, lihat Keamanan dan Izin Data Wrangler di Panduan Pengembang Amazon AI. SageMaker Untuk informasi selengkapnya tentang peran SageMaker AI, lihat Peran SageMaker AI.

  Untuk memberikan akses ke operasi impor data Amazon Personalize, lampirkan kebijakan IAM berikut ke peran eksekusi SageMaker AI. Kebijakan ini memberikan izin yang diperlukan untuk mengimpor data ke Amazon Personalisasi dan melampirkan kebijakan ke bucket Amazon S3 Anda. Dan itu memberikan `PassRole` izin saat layanannya Amazon Personalisasi. Perbarui Amazon S3 `amzn-s3-demo-bucket` ke nama bucket Amazon S3 yang ingin Anda gunakan sebagai tujuan untuk data yang diformat setelah Anda menyiapkannya dengan Data Wrangler.

  JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "personalize:Create*",
                "personalize:List*",
                "personalize:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutBucketPolicy"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
```

```
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "personalize.amazonaws.com"
                }
            }
        }
    ]
}
```

Untuk informasi tentang cara membuat kebijakan IAM, lihat Membuat kebijakan IAM di Panduan Pengguna IAM. Untuk informasi tentang melampirkan kebijakan IAM ke peran, lihat Menambahkan dan menghapus izin identitas IAM di Panduan Pengguna IAM.

Meluncurkan Data Wrangler dari Amazon Personalize

Untuk meluncurkan Data Wrangler dari Amazon Personalize, Anda SageMaker menggunakan konsol Amazon Personalize untuk mengonfigurasi domain AI dan meluncurkan Data Wrangler.

Untuk meluncurkan Data Wrangler dari Amazon Personalize

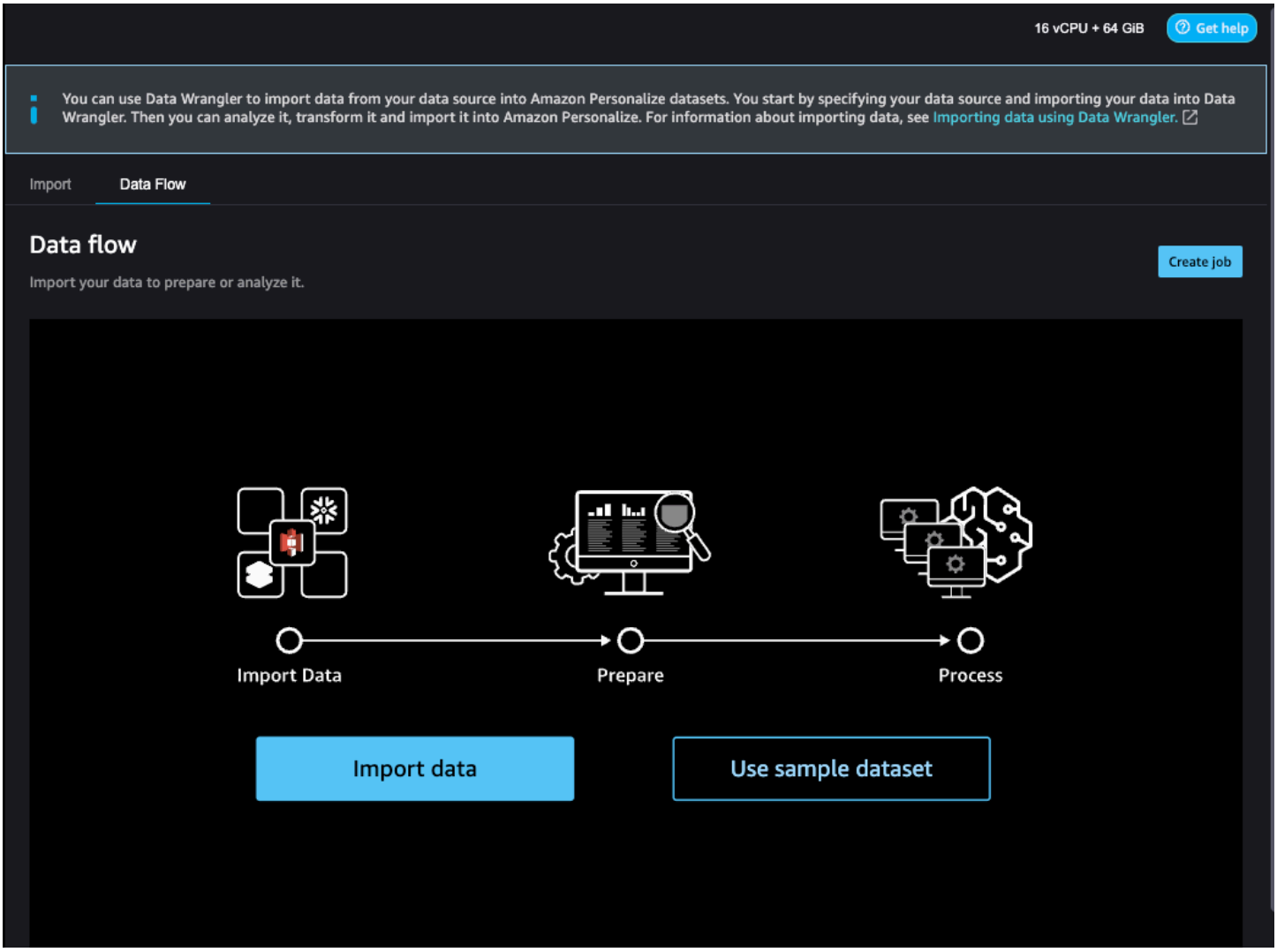
1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
3. Di Siapkan kumpulan data pilih Buat kumpulan data dan pilih jenis kumpulan data yang akan dibuat. Anda tidak dapat menggunakan Data Wrangler untuk menyiapkan kumpulan data Tindakan atau kumpulan data interaksi Tindakan.
4. Pilih Impor data menggunakan Data Wrangler dan pilih Berikutnya.
5. Untuk SageMaker domain, pilih untuk menggunakan domain yang ada atau buat yang baru. Anda memerlukan Domain SageMaker AI untuk mengakses Data Wrangler di SageMaker AI Studio Classic. Untuk informasi tentang domain dan profil pengguna, lihat Domain SageMaker AI di Panduan Pengembang Amazon SageMaker AI.

6. Untuk menggunakan domain yang ada, pilih domain SageMaker AI dan profil Pengguna untuk mengonfigurasi domain.
7. Untuk membuat domain baru:
   - Berikan nama domain baru.
   - Pilih nama profil pengguna.
   - Untuk peran Eksekusi, pilih peran yang Anda buatMenyiapkan izin. Atau, jika Anda memiliki CreateRole izin, buat peran baru menggunakan panduan pembuatan peran. Peran yang Anda gunakan harus memiliki `AmazonSageMakerFullAccess` kebijakan yang terlampir.
8. Pilih Berikutnya. Jika Anda membuat domain baru, SageMaker AI mulai membuat domain Anda. Ini bisa memakan waktu hingga sepuluh menit.
9. Tinjau detail untuk domain SageMaker AI Anda.
10. Pilih Impor data dengan Data Wrangler. SageMaker AI Studio Classic mulai menciptakan lingkungan Anda, dan ketika selesai, halaman aliran Data Wrangler di SageMaker AI Studio Classic terbuka di tab baru. Diperlukan waktu hingga lima menit bagi SageMaker AI Studio Classic untuk menyelesaikan pembuatan lingkungan Anda. Setelah selesai, Anda siap untuk mulai mengimpor data ke Data Wrangler. Untuk informasi selengkapnya, lihat Mengimpor data ke Data Wrangler.

Mengimpor data ke Data Wrangler

Setelah mengonfigurasi domain SageMaker AI dan meluncurkan Data Wrangler di tab baru, Anda siap untuk mengimpor data dari sumber Anda ke Data Wrangler. Saat Anda menggunakan Data Wrangler untuk menyiapkan data untuk Amazon Personalize, Anda mengimpor satu kumpulan data pada satu waktu. Kami merekomendasikan memulai dengan kumpulan data interaksi Item. Anda tidak dapat menggunakan Data Wrangler untuk menyiapkan kumpulan data Tindakan atau kumpulan data interaksi Tindakan.

Anda mulai pada halaman Aliran data. Halaman akan terlihat mirip dengan yang berikut ini.



Untuk mulai mengimpor data, Anda memilih Impor data dan menentukan sumber data Anda. Data Wrangler mendukung 40+ sumber. Ini termasuk AWS layanan, seperti Amazon Redshift, Amazon EMR, atau Amazon Athena, dan pihak ketiga, seperti Snowflake atau. DataBricks Sumber data yang berbeda memiliki prosedur yang berbeda untuk menghubungkan dan mengimpor data.

Untuk daftar lengkap sumber dan step-by-step petunjuk yang tersedia tentang mengimpor data, lihat Mengimpor di Panduan Pengembang Amazon SageMaker AI.

Setelah Anda mengimpor data ke Data Wrangler, Anda siap untuk mengubahnya. Untuk informasi tentang mengubah data, lihatMengubah data.

Mengubah data

Untuk mengubah data di Data Wrangler, Anda menambahkan langkah Transform ke aliran data Anda. Data Wrangler mencakup lebih dari 300 transformasi yang dapat Anda gunakan untuk

menyiapkan data, termasuk kolom Peta untuk transformasi Amazon Personalize. Dan Anda dapat menggunakan transformasi Data Wrangler umum untuk memperbaiki masalah seperti outlier, masalah tipe, dan nilai yang hilang.

Setelah Anda selesai mengubah data Anda, Anda dapat menganalisisnya dengan Data Wrangler. Atau, jika Anda selesai menyiapkan data Anda di Data Wrangler, Anda dapat memprosesnya dan mengimpornya ke Amazon Personalize. Untuk informasi tentang menganalisis data, lihatMenghasilkan visualisasi dan wawasan data. Untuk informasi tentang memproses dan mengimpor data, lihatMemproses data dan mengimpornya ke Amazon Personalize.

Topik

- Kolom pemetaan untuk Amazon Personalisasi
- Data Umum Wrangler berubah

Kolom pemetaan untuk Amazon Personalisasi

Untuk mengubah data Anda sehingga memenuhi persyaratan Amazon Personalize, Anda menambahkan kolom Peta untuk Amazon Personalize mengubah dan memetakan kolom Anda ke bidang wajib dan opsional untuk Amazon Personalize.

Untuk menggunakan kolom Peta untuk Amazon Personalize transform

1. Pilih + untuk transformasi terbaru Anda dan pilih Tambahkan transformasi. Jika Anda belum menambahkan transformasi, pilih + untuk transformasi tipe Data. Data Wrangler menambahkan transformasi ini secara otomatis ke alur Anda.
2. Pilih Tambahkan langkah.
3. Pilih Transformasi untuk Amazon Personalisasi. Kolom Peta untuk transformasi Amazon Personalize dipilih secara default.
4. Gunakan bidang transformasi untuk memetakan data Anda ke atribut Amazon Personalize yang diperlukan.

    1. Pilih jenis kumpulan data yang cocok dengan data Anda (Interaksi, Item, atau Pengguna).
    2. Pilih domain Anda (ECOMMERCE, VIDEO_ON_DEMAND, atau kustom). Domain yang Anda pilih harus sesuai dengan domain yang Anda tentukan saat membuat grup kumpulan data.
    3. Pilih kolom yang cocok dengan bidang wajib dan opsional untuk Amazon Personalize. Misalnya, untuk kolom Item_ID, pilih kolom dalam data Anda yang menyimpan informasi identifikasi unik untuk setiap item Anda.

Setiap kolom kolom disaring berdasarkan tipe data. Hanya kolom dalam data Anda yang memenuhi persyaratan tipe data Amazon Personalize yang tersedia. Jika data Anda bukan tipe yang diperlukan, Anda dapat menggunakan Transformasi Parse Value as Type Data Wrangler untuk mengonversinya.

Data Umum Wrangler berubah

Transformasi Data Wrangler umum berikut dapat membantu Anda menyiapkan data untuk Amazon Personalize:

- Konversi tipe data: Jika bidang Anda tidak terdaftar sebagai opsi yang memungkinkan di kolom Peta untuk transformasi Amazon Personalisasi, Anda mungkin perlu mengonversi tipe datanya. Data Wrangler mengubah Nilai Parse sebagai Jenis dapat membantu Anda mengonversi data Anda. Atau Anda dapat menggunakan transformasi tipe Data yang ditambahkan Data Wrangler secara default saat Anda membuat alur. Untuk menggunakan transformasi ini, Anda memilih tipe data dari daftar drop-down Jenis, pilih Pratinjau dan kemudian pilih Perbarui.

  Untuk informasi tentang tipe data yang diperlukan untuk bidang, lihat bagian untuk jenis domain dan kumpulan data Anda. Membuat file JSON skema untuk skema Amazon Personalize
- Menangani nilai dan outlier yang hilang: Jika Anda menghasilkan nilai yang hilang atau wawasan outlier, Anda dapat menggunakan Transformasi Data Wrangler Menangani Pencilan dan Menangani Nilai yang Hilang untuk mengatasi masalah ini.
- Transformasi kustom: Dengan Data Wrangler, Anda dapat membuat transformasi Anda sendiri dengan Python (User-Defined Function) PySpark,, pandas, atau (SQL). PySpark Anda dapat menggunakan transformasi khusus untuk melakukan tugas seperti menjatuhkan kolom duplikat atau pengelompokan berdasarkan kolom. Untuk informasi selengkapnya, lihat Transformasi Kustom di Panduan Pengembang Amazon SageMaker AI.

Menghasilkan visualisasi dan wawasan data

Setelah mengimpor data ke Data Wrangler, Anda dapat menggunakannya untuk menghasilkan visualisasi dan wawasan data.

- Visualisasi: Data Wrangler dapat menghasilkan berbagai jenis grafik, seperti histogram dan plot pencar. Misalnya, Anda dapat membuat histogram untuk mengidentifikasi outlier dalam data Anda.

- Wawasan data: Anda dapat menggunakan Laporan Kualitas Data dan Wawasan untuk Amazon Personalisasi untuk mempelajari data Anda melalui wawasan data serta statistik kolom dan baris. Laporan ini dapat memberi tahu Anda jika Anda memiliki masalah jenis apa pun dalam data Anda. Dan Anda dapat mempelajari tindakan apa yang dapat Anda ambil untuk meningkatkan data Anda. Tindakan ini dapat membantu Anda memenuhi persyaratan sumber daya Amazon Personalize, seperti persyaratan pelatihan model, atau dapat mengarah pada rekomendasi yang lebih baik.

Setelah mempelajari data Anda melalui visualisasi dan wawasan, Anda dapat menggunakan informasi ini untuk membantu Anda menerapkan transformasi tambahan untuk meningkatkan data Anda. Atau, jika Anda selesai menyiapkan data, Anda dapat memprosesnya dan mengimpornya ke Amazon Personalize. Untuk informasi tentang mengubah data Anda, lihatMengubah data. Untuk informasi tentang memproses dan mengimpor data, lihatMemproses data dan mengimpornya ke Amazon Personalize.

Menghasilkan visualisasi

Anda dapat menggunakan Data Wrangler untuk membuat berbagai jenis grafik, seperti histogram dan plot pencar. Misalnya, Anda dapat membuat histogram untuk mengidentifikasi outlier dalam data Anda. Untuk menghasilkan visualisasi data, Anda menambahkan langkah Analisis ke alur Anda dan, dari tipe Analisis, pilih visualisasi yang ingin Anda buat.

Untuk informasi selengkapnya tentang membuat visualisasi di Data Wrangler, lihat Menganalisis dan Memvisualisasikan di Panduan Pengembang Amazon SageMaker AI.

Menghasilkan wawasan data

Anda dapat menggunakan Data Wrangler untuk menghasilkan laporan Kualitas Data dan Wawasan untuk Amazon Personalisasi khusus untuk jenis kumpulan data Anda. Sebelum membuat laporan, kami menyarankan Anda mengubah data Anda untuk memenuhi persyaratan Amazon Personalize. Ini akan menghasilkan wawasan yang lebih relevan. Untuk informasi selengkapnya, lihat Mengubah data.

Topik

- Melaporkan konten
- Menghasilkan laporan

Melaporkan konten

Laporan Kualitas Data dan Wawasan untuk Amazon Personalize mencakup bagian berikut:

- Ringkasan: Ringkasan laporan mencakup statistik kumpulan data dan peringatan prioritas tinggi:
  - Statistik kumpulan data: Ini termasuk Amazon Personalisasi statistik tertentu, seperti jumlah pengguna unik dalam data interaksi Anda, dan statistik umum, seperti jumlah nilai atau outlier yang hilang.
  - Peringatan prioritas tinggi: Ini adalah Amazon Personalisasi wawasan spesifik yang memiliki dampak paling besar pada pelatihan atau rekomendasi. Setiap peringatan mencakup tindakan yang disarankan yang dapat Anda ambil untuk menyelesaikan masalah.
- Baris duplikat dan Baris tidak lengkap: Bagian ini mencakup informasi tentang baris mana yang memiliki nilai yang hilang dan baris mana yang diduplikasi dalam data Anda.
- Ringkasan fitur: Bagian ini mencakup tipe data untuk setiap kolom, informasi data yang tidak valid atau hilang, dan jumlah peringatan.
- Detail fitur: Bagian ini mencakup subbagian dengan informasi terperinci untuk setiap kolom data Anda. Setiap subbagian mencakup statistik untuk kolom, seperti jumlah nilai kategoris, dan informasi nilai yang hilang. Dan setiap subbagian menyertakan Amazon Personalisasi wawasan spesifik dan tindakan yang disarankan untuk kolom data. Misalnya, wawasan mungkin menunjukkan bahwa kolom memiliki lebih dari 30 kategori yang mungkin.

Masalah tipe data

Laporan mengidentifikasi kolom yang bukan tipe data yang benar dan menentukan jenis yang diperlukan. Untuk mendapatkan wawasan yang terkait dengan fitur-fitur ini, Anda harus mengonversi tipe data kolom dan membuat laporan lagi. Untuk mengonversi tipe, Anda dapat menggunakan Data Wrangler mengubah Nilai Parse sebagai Jenis.

Amazon Personalisasi wawasan

Wawasan Personalisasi Amazon mencakup temuan dan tindakan yang disarankan. Tindakan ini opsional. Misalnya, laporan mungkin menyertakan wawasan dan tindakan yang terkait dengan jumlah kategori untuk kolom data kategoris. Jika Anda tidak percaya kolom itu kategoris, Anda dapat mengabaikan wawasan ini dan tidak mengambil tindakan.

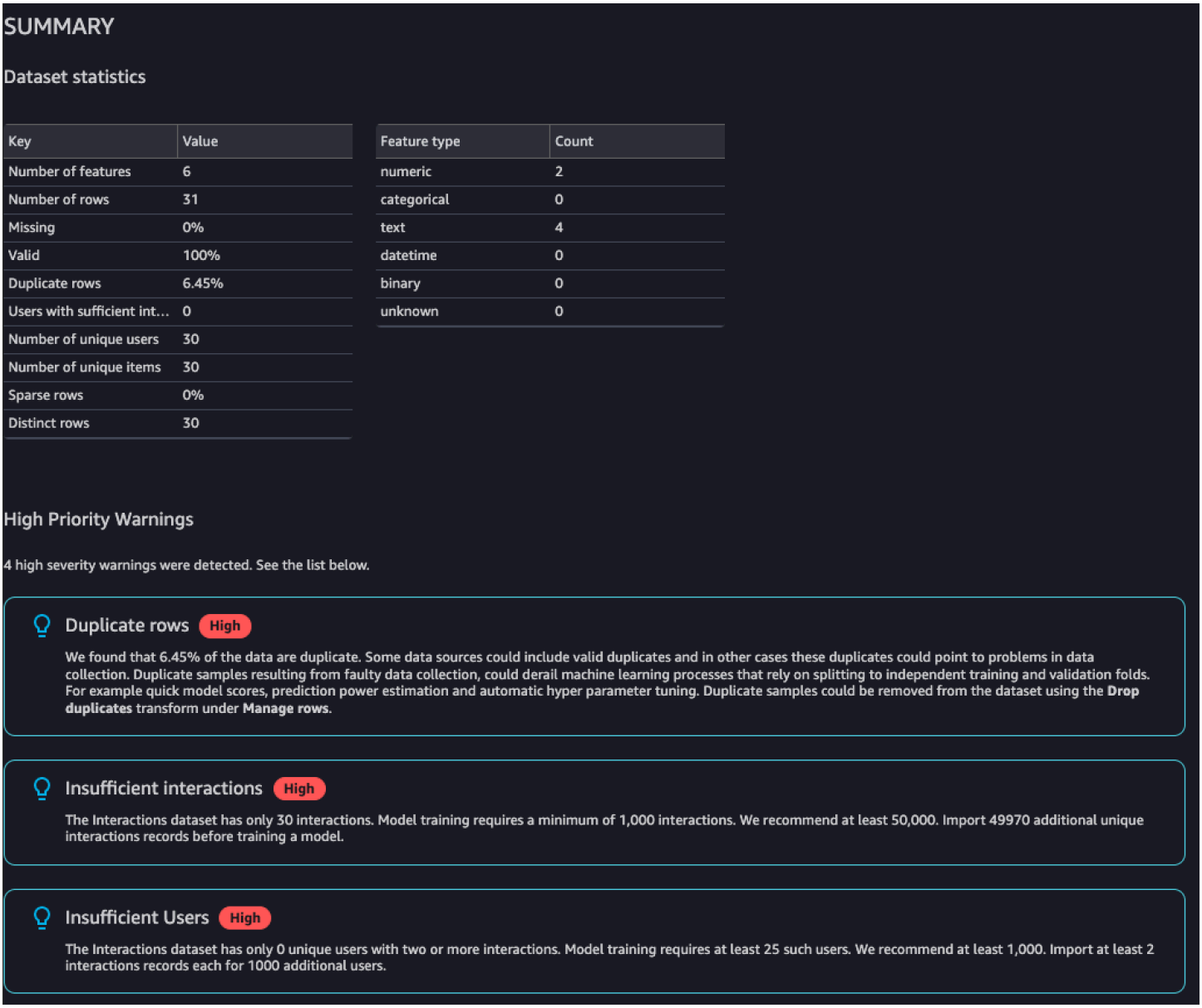
Kecuali untuk perbedaan kata-kata kecil, Amazon Personalisasi wawasan spesifik sama dengan wawasan kumpulan data tunggal yang mungkin Anda hasilkan saat menganalisis data Anda dengan

Amazon Personalize. Misalnya, laporan wawasan di Data Wrangler mencakup wawasan seperti "Kumpulan data interaksi Item hanya memiliki X pengguna unik dengan dua atau lebih interaksi." Tapi itu tidak termasuk wawasan seperti "X% item dalam kumpulan data Item tidak memiliki interaksi dalam kumpulan data interaksi Item."

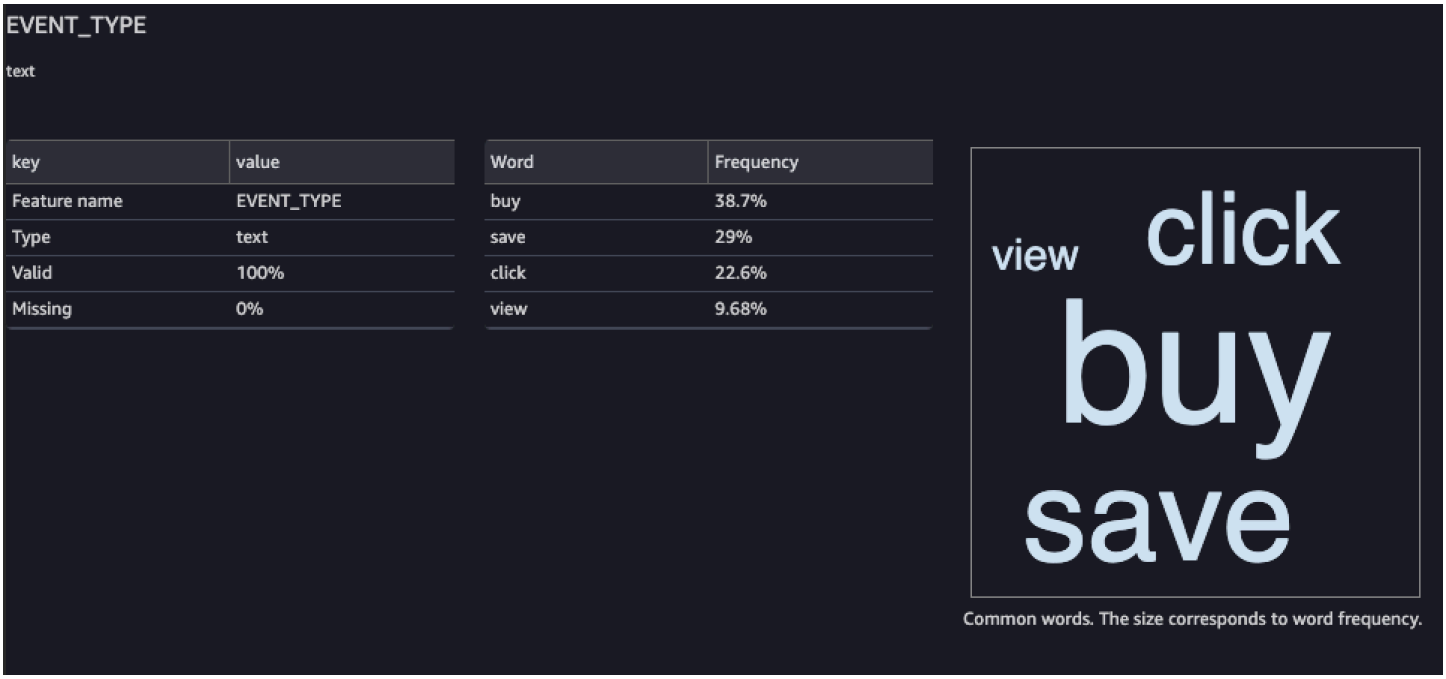
Untuk daftar kemungkinan Amazon Personalisasi wawasan tertentu, lihat wawasan yang tidak mereferensikan beberapa kumpulan data. Wawasan data

Contoh laporan

Tampilan dan nuansa laporan Amazon Personalize sama dengan laporan wawasan umum di Data Wrangler. Untuk contoh laporan wawasan umum, lihat Mendapatkan Wawasan Tentang Data dan Kualitas Data di Panduan Pengembang Amazon SageMaker AI. Contoh berikut menunjukkan bagaimana bagian ringkasan laporan untuk kumpulan data interaksi Item. Ini mencakup statistik kumpulan data dan beberapa kemungkinan peringatan kumpulan data interaksi Item prioritas tinggi.



Contoh berikut menunjukkan bagaimana bagian detail fitur untuk kolom EVENT_TYPE dari kumpulan data interaksi Item mungkin muncul dalam laporan.



Menghasilkan laporan

Untuk menghasilkan Laporan Kualitas Data dan Wawasan untuk Amazon Personalisasi, pilih Dapatkan wawasan data untuk transformasi dan buat analisis.

Untuk menghasilkan Laporan Kualitas Data dan Wawasan untuk Amazon Personalisasi

1. Pilih opsi + untuk transformasi yang Anda analisis. Jika Anda belum menambahkan transformasi, pilih + untuk transformasi tipe Data. Data Wrangler menambahkan transformasi ini secara otomatis ke alur Anda.
2. Pilih Dapatkan wawasan data. Panel Create analysis akan ditampilkan.
3. Untuk jenis Analisis, pilih Laporan Kualitas Data dan Wawasan untuk Amazon Personalisasi.
4. Untuk jenis Dataset, pilih jenis kumpulan data Amazon Personalize yang Anda analisis.
5. Secara opsional pilih Jalankan pada data lengkap. Secara default, Data Wrangler menghasilkan wawasan hanya pada sampel data Anda.
6. Pilih Buat. Ketika analisis selesai, laporan muncul.

Memproses data dan mengimpornya ke Amazon Personalize

Ketika Anda selesai menganalisis dan mengubah data Anda, Anda siap untuk memprosesnya dan mengimpornya ke Amazon Personalize.

- Memproses data — Memproses data menerapkan transformasi Anda ke seluruh kumpulan data Anda dan mengeluarkannya ke tujuan yang Anda tentukan. Dalam hal ini Anda menentukan bucket Amazon S3.
- Mengimpor data ke Amazon Personalize — Untuk mengimpor data yang diproses ke Amazon Personalize, Anda menjalankan Notebook Jupyter yang disediakan di AI Studio Classic. SageMaker Notebook ini membuat kumpulan data Amazon Personalize Anda dan mengimpor data Anda ke dalamnya.

Pemrosesan data

Sebelum mengimpor data ke Amazon Personalize, Anda harus menerapkan transformasi ke seluruh kumpulan data dan mengeluarkannya ke bucket Amazon S3. Untuk melakukannya, Anda membuat node tujuan dengan tujuan disetel ke bucket Amazon S3, lalu meluncurkan pekerjaan pemrosesan untuk transformasi.

Untuk step-by-step petunjuk tentang menentukan tujuan dan meluncurkan pekerjaan proses, lihat Meluncurkan pekerjaan pemrosesan dengan beberapa klik menggunakan Amazon SageMaker AI Data Wrangler. Saat Anda menambahkan tujuan, pilih Amazon S3. Anda akan menggunakan lokasi ini saat mengimpor data yang diproses ke Amazon Personalize.

Setelah selesai memproses data, Anda siap mengimpornya dari bucket Amazon S3 ke Amazon Personalize.

Mengimpor data ke Amazon Personalize

Setelah Anda memproses data Anda, Anda siap untuk mengimpornya ke Amazon Personalize. Untuk mengimpor data yang diproses ke Amazon Personalize, Anda menjalankan Notebook Jupyter yang disediakan di SageMaker AI Studio Classic. Notebook ini membuat kumpulan data Amazon Personalize Anda dan mengimpor data Anda ke dalamnya.

Untuk mengimpor data yang diproses ke Amazon Personalize

1. Untuk transformasi yang ingin Anda ekspor, pilih Ekspor ke dan pilih Amazon Personalize (via Jupyter Notebook).
2. Ubah buku catatan untuk menentukan bucket Amazon S3 yang Anda gunakan sebagai tujuan data untuk pekerjaan pemrosesan. Secara opsional tentukan domain untuk grup kumpulan data Anda. Secara default, notebook membuat grup dataset kustom.

3. Tinjau sel notebook yang membuat skema. Verifikasi bahwa bidang skema memiliki tipe dan atribut yang diharapkan sebelum menjalankan sel.

   - Verifikasi bahwa bidang yang mendukung data nol telah `null` tercantum dalam daftar jenis. Contoh berikut menunjukkan cara menambahkan `null` untuk bidang.

   ```
   {
     "name": "GENDER",
     "type": [
       "null",
       "string"
     ],
     "categorical": true
   }
   ```

   - Verifikasi bahwa bidang kategoris memiliki atribut kategoris disetel ke true. Contoh berikut menunjukkan cara menandai bidang kategoris.

   ```
   {
             "name": "SUBSCRIPTION_MODEL",
             "type": "string",
             "categorical": true
   }
   ```

   - Verifikasi bahwa bidang tekstual memiliki atribut tekstual disetel ke true. Contoh berikut menunjukkan cara menandai bidang sebagai tekstual.

   ```
   {
         "name": "DESCRIPTION",
         "type": [
           "null",
           "string"
         ],
         "textual": true
   }
   ```

4. Jalankan buku catatan untuk membuat skema, dan membuat kumpulan data, dan impor data Anda ke dalam kumpulan data Amazon Personalize. Anda menjalankan notebook seperti halnya notebook di luar SageMaker AI Studio Classic. Untuk informasi tentang menjalankan notebook Jupyter, lihat Menjalankan Kode. Untuk informasi tentang notebook di SageMaker AI Studio

Classic, lihat Menggunakan Notebook Amazon SageMaker AI di Panduan Pengembang Amazon SageMaker AI.

Setelah Anda menyelesaikan buku catatan, jika Anda mengimpor data interaksi, Anda siap untuk membuat pemberi rekomendasi atau sumber daya khusus. Atau Anda dapat mengulangi prosesnya dengan kumpulan data item atau kumpulan data pengguna.

- Untuk informasi tentang membuat pemberi rekomendasi domain, lihatRekomendasi domain di Amazon Personalisasi.
- Untuk informasi tentang membuat dan menerapkan sumber daya kustom, lihatSumber daya khusus untuk melatih dan menerapkan model Amazon Personalize.

Mengimpor catatan individual ke dalam kumpulan data Amazon Personalize

Setelah selesaiMembuat skema dan dataset, Anda dapat mengimpor catatan individual, termasuk interaksi item, pengguna, item, tindakan, atau interaksi tindakan ke dalam kumpulan data yang ada. Mengimpor data satu per satu memungkinkan Anda menambahkan sejumlah kecil catatan ke kumpulan data Amazon Personalisasi saat katalog Anda bertambah. Anda dapat mengimpor hingga 10 catatan per operasi impor individual.

Jika Anda mengimpor item, pengguna, atau tindakan dengan ID yang sama dengan catatan yang sudah ada di kumpulan data Anda, Amazon Personalize menggantinya dengan catatan baru. Jika Anda merekam dua interaksi item atau peristiwa interaksi tindakan dengan stempel waktu yang sama persis dan properti yang identik, Amazon Personalize hanya menyimpan satu peristiwa.

Jika Anda menggunakan Apache Kafka, Anda dapat menggunakan konektor Kafka untuk Amazon Personalize untuk mengalirkan data secara real time ke Amazon Personalize. Untuk informasi, lihat Konektor Kafka untuk Amazon Personalisasi di repositori Github personalize-kafka-connector.

Jika Anda memiliki sejumlah besar catatan sejarah, kami sarankan Anda terlebih dahulu mengimpor data secara massal dan kemudian mengimpor data satu per satu sesuai kebutuhan. Lihat Mengimpor data massal ke Amazon Personalisasi dengan pekerjaan impor kumpulan data.

Filter pembaruan untuk impor catatan individual

Amazon Personalize memperbarui filter apa pun yang Anda buat di grup kumpulan data dengan interaksi, item, dan data pengguna baru Anda dalam waktu 20 menit dari impor individual terakhir.

Pembaruan ini memungkinkan kampanye Anda untuk menggunakan data terbaru Anda saat memfilter rekomendasi untuk pengguna Anda.

Jika Anda telah membuat pemberi rekomendasi atau menerapkan versi solusi kustom dengan kampanye, bagaimana catatan individu baru memengaruhi rekomendasi bergantung pada kasus penggunaan domain atau resep yang Anda gunakan. Lihat informasi yang lebih lengkap di Memperbarui data dalam kumpulan data setelah pelatihan.

Topik

- Mengimpor interaksi secara individual
- Mengimpor pengguna secara individual
- Mengimpor barang secara individual
- Mengimpor tindakan secara individual

Mengimpor interaksi secara individual

Setelah Anda selesai Membuat skema dan dataset membuat kumpulan data interaksi Item, Anda dapat mengimpor satu atau lebih peristiwa baru ke dalam kumpulan data secara individual. Untuk mengimpor peristiwa interaksi satu per satu, Anda membuat pelacak peristiwa dan kemudian mengimpor satu atau beberapa peristiwa ke dalam kumpulan data interaksi Item Anda. Anda dapat mengimpor peristiwa interaksi individual historis menggunakan konsol Amazon Personalize, atau mengimpor peristiwa historis atau real-time menggunakan AWS Command Line Interface (AWS CLI), atau. AWS SDKs

Bagian ini mencakup informasi tentang mengimpor acara dengan konsol Amazon Personalize. Sebaiknya gunakan konsol Amazon Personalize untuk mengimpor hanya peristiwa historis. Untuk informasi tentang menggunakan AWS CLI atau AWS SDKs untuk merekam peristiwa secara real-time, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi.

Untuk informasi tentang cara Amazon Personalisasi pembaruan memfilter untuk catatan baru dan bagaimana catatan baru memengaruhi rekomendasi, lihat. Mengimpor catatan individual ke dalam kumpulan data Amazon Personalize

Topik

- Membuat pelacak acara (konsol)
- Mengimpor acara satu per satu (konsol)

Membuat pelacak acara (konsol)

> **Note**
>
> Jika Anda telah membuat pelacak acara, Anda dapat melompat keMengimpor acara satu per satu (konsol).

Sebelum dapat mengimpor peristiwa ke kumpulan data Interaksi, Anda harus membuat pelacak peristiwa untuk grup kumpulan data.

Untuk membuat pelacak acara (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Di halaman Grup kumpulan data, pilih grup kumpulan data dengan Kumpulan data interaksi item yang ingin Anda impor acara.
3. Di Dasbor untuk grup kumpulan data, di Instal event ingestion SDK, pilih Mulai.
4. Pada halaman Configure tracker, di konfigurasi Tracker, untuk nama Tracker, berikan nama untuk pelacak peristiwa, dan pilih Berikutnya.
5. Halaman Instal SDK menampilkan ID Pelacakan untuk pelacak peristiwa baru dan instruksi untuk menggunakan AWS Amplify atau mengalirkan AWS Lambda data peristiwa.

   Anda dapat mengabaikan informasi ini karena Anda menggunakan konsol Amazon Personalize untuk mengunggah data peristiwa. Jika Anda ingin melakukan streaming data acara menggunakan AWS Amplify atau AWS Lambda di masa depan, Anda dapat melihat informasi ini dengan memilih pelacak peristiwa di halaman Pelacak acara.
6. Pilih Selesai. Anda sekarang dapat mengimpor peristiwa dengan konsol (melihat Mengimpor acara satu per satu (konsol) atau merekam peristiwa secara real time menggunakan `PutEvents` operasi (lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi).

Mengimpor acara satu per satu (konsol)

Setelah Anda membuat pelacak peristiwa, Anda dapat mengimpor peristiwa satu per satu ke dalam kumpulan data interaksi Item. Prosedur ini mengasumsikan Anda telah membuat kumpulan data interaksi Item. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Untuk mengimpor acara satu per satu (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Di halaman Grup kumpulan data, pilih grup kumpulan data dengan Kumpulan data interaksi item yang ingin Anda impor acara.
3. Di panel navigasi, pilih kumpulan data.
4. Pada halaman Datasets, pilih Dataset Interaksi.
5. Di kanan atas halaman detail kumpulan data, pilih Ubah kumpulan data, dan pilih Buat catatan.
6. Di Buat catatan interaksi item pengguna halaman, untuk Rekam masukan, masukkan detail acara dalam format JSON. Nama dan nilai bidang acara harus cocok dengan skema yang Anda gunakan saat membuat kumpulan data interaksi Item. Amazon Personalize menyediakan template JSON dengan nama bidang dan tipe data dari skema ini. Anda dapat mengimpor hingga 10 acara sekaligus.
7. Pilih Buat catatan (s). Sebagai Respons, hasil impor terdaftar dan pesan sukses atau gagal ditampilkan.

Mengimpor pengguna secara individual

Setelah selesai Membuat skema dan dataset membuat kumpulan data Pengguna, Anda dapat mengimpor satu atau beberapa pengguna baru secara individual ke dalam kumpulan data. Pengguna yang mengimpor secara individual memungkinkan Anda untuk menjaga kumpulan data Pengguna tetap terkini dengan impor batch kecil seiring pertumbuhan katalog Anda. Anda dapat mengimpor hingga 10 pengguna sekaligus. Jika Anda memiliki sejumlah besar pengguna baru, kami sarankan Anda terlebih dahulu mengimpor data secara massal dan kemudian mengimpor data pengguna satu per satu sesuai kebutuhan. Lihat Mengimpor data massal ke Amazon Personalisasi dengan pekerjaan impor kumpulan data.

Anda dapat menggunakan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau AWS SDKs untuk mengimpor pengguna. Jika Anda mengimpor pengguna yang `userId` sama dengan pengguna yang sudah ada di kumpulan data Pengguna, Amazon Personalize akan menggantikan pengguna dengan yang baru. Anda dapat mengimpor hingga 10 pengguna sekaligus.

Untuk informasi tentang cara Amazon Personalisasi pembaruan memfilter untuk catatan baru dan bagaimana catatan baru memengaruhi rekomendasi, lihat. Mengimpor catatan individual ke dalam kumpulan data Amazon Personalize

Topik

- Mengimpor pengguna secara individual (konsol)
- Mengimpor pengguna secara individual ()AWS CLI
- Mengimpor pengguna secara individual ()AWS SDKs

Mengimpor pengguna secara individual (konsol)

Anda dapat mengimpor hingga 10 pengguna sekaligus. Prosedur ini mengasumsikan Anda telah membuat kumpulan data Pengguna. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Untuk mengimpor pengguna satu per satu (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data dengan kumpulan data Pengguna yang ingin Anda impor pengguna.
3. Di panel navigasi, pilih Datasets.
4. Pada halaman Datasets, pilih Dataset Pengguna.
5. Pada halaman detail kumpulan data, di kanan atas, pilih Ubah kumpulan data dan pilih Buat catatan.
6. Pada halaman Buat catatan pengguna, untuk input rekaman, masukkan detail pengguna dalam format JSON. Nama dan nilai bidang pengguna harus sesuai dengan skema yang Anda gunakan saat membuat kumpulan data Pengguna. Amazon Personalize menyediakan template JSON dengan nama bidang dan tipe data dari skema ini.
7. Pilih Buat catatan. Sebagai Respons, hasil impor terdaftar dan pesan sukses atau gagal ditampilkan.

Mengimpor pengguna secara individual ()AWS CLI

Tambahkan satu atau beberapa pengguna ke kumpulan data Pengguna Anda dengan PutUsers operasi. Anda dapat mengimpor hingga 10 pengguna dengan satu `PutUsers` panggilan. Bagian ini mengasumsikan bahwa Anda telah membuat kumpulan data Pengguna. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Gunakan `put-users` perintah berikut untuk menambahkan satu atau lebih pengguna dengan file AWS CLI. Ganti `dataset arn` dengan Nama Sumber Daya Amazon (ARN) dari kumpulan data Anda dan `user Id` dengan ID pengguna. Jika pengguna dengan hal yang sama sudah `userId` ada di kumpulan data Pengguna Anda, Amazon Personalize menggantinya dengan yang baru.

Untuk`properties`, untuk setiap bidang dalam kumpulan data Pengguna Anda, ganti `propertyName` dengan nama bidang dari skema Anda dalam kasus unta. Misalnya, GENDER akan menjadi `gender` dan MEMBERSHIP_TYPE akan menjadi. `membershipType` Ganti `user data` dengan data untuk pengguna. Untuk data string kategoris, untuk menyertakan beberapa kategori untuk satu properti, pisahkan setiap kategori dengan pipe ()`|`. Sebagai contoh, `\"Premium Class|Legacy Member\"`.

```
aws personalize-events put-users \
  --dataset-arn dataset arn \
  --users '[{
      "userId": "user Id",
      "properties": "{\"propertyName\": "\user data\"}"
    },
    {
      "userId": "user Id",
      "properties": "{\"propertyName\": "\user data\"}"
    }]'
```

Mengimpor pengguna secara individual ()AWS SDKs

Tambahkan satu atau beberapa pengguna ke kumpulan data Pengguna Anda dengan PutUsers operasi. Jika pengguna dengan hal yang sama sudah `userId` ada di kumpulan data Pengguna Anda, Amazon Personalize menggantinya dengan yang baru. Anda dapat mengimpor hingga 10 pengguna dengan satu `PutUsers` panggilan. Bagian ini mengasumsikan bahwa Anda telah membuat kumpulan data Pengguna. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Kode berikut menunjukkan cara menambahkan satu atau beberapa pengguna ke kumpulan data Pengguna Anda. Untuk setiap parameter nama properti, berikan nama bidang dari skema Anda dalam kasus unta. Misalnya, GENDER akan menjadi `gender` dan MEMBERSHIP_TYPE akan menjadi. `membershipType` Untuk setiap parameter nilai properti, berikan data untuk pengguna.

Untuk data string kategoris, untuk menyertakan beberapa kategori untuk satu properti pisahkan setiap kategori dengan pipe ()`|`. Sebagai contoh, `"Premium class|Legacy Member"`.

SDK for Python (Boto3)

```
import boto3

personalize_events = boto3.client(service_name='personalize-events')

personalize_events.put_users(
    datasetArn = 'dataset arn',
    users = [{
      'userId': 'user ID',
      'properties': "{\"propertyName\": \"user data\"}"
      },
      {
      'userId': 'user ID',
      'properties': "{\"propertyName\": \"user data\"}"
      }]
)
```

SDK for Java 2.x

```
public static int putUsers(PersonalizeEventsClient personalizeEventsClient,
                           String datasetArn,
                           String user1Id,
                           String user1PropertyName,
                           String user1PropertyValue,
                           String user2Id,
                           String user2PropertyName,
                           String user2PropertyValue) {

    int responseCode = 0;
    ArrayList<User> users = new ArrayList<>();

    try {
        User user1 = User.builder()
          .userId(user1Id)
          .properties(String.format("{\"%1$s\": \"%2$s\"}", user1PropertyName,
 user1PropertyValue))
          .build();

        users.add(user1);

        User user2 = User.builder()
          .userId(user2Id)
```

```
          .properties(String.format("{\"%1$s\": \"%2$s\"}", user2PropertyName,
 user2PropertyValue))
          .build();

        users.add(user2);

        PutUsersRequest putUsersRequest = PutUsersRequest.builder()
          .datasetArn(datasetArn)
          .build();

        responseCode =
 personalizeEventsClient.putUsers(putUsersRequest).sdkHttpResponse().statusCode();
        System.out.println("Response code: " + responseCode);
        return responseCode;

    } catch (PersonalizeEventsException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return responseCode;
}
```

SDK for JavaScript v3

```
import {
  PutUsersCommand,
  PersonalizeEventsClient,
} from "@aws-sdk/client-personalize-events";

const personalizeEventsClient = new PersonalizeEventsClient({
  region: "REGION",
});

// set the put users parameters
var putUsersParam = {
  datasetArn:
    "DATASET ARN",
  users: [
    {
      userId: "userId",
      properties: '{"column1Name": "value", "column2Name": "value"}',
    },
    {
      userId: "userId",
```

```
      properties: '{"column1Name": "value", "column2Name": "value"}',
    },
  ],
};
export const run = async () => {
  try {
    const response = await personalizeEventsClient.send(
      new PutUsersCommand(putUsersParam)
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Mengimpor barang secara individual

Setelah selesai Membuat skema dan dataset membuat kumpulan data Item, Anda dapat mengimpor satu atau beberapa item baru ke dalam kumpulan data secara individual. Mengimpor item secara individual memungkinkan Anda untuk menjaga dataset Item tetap terkini dengan impor batch kecil seiring pertumbuhan katalog Anda. Anda dapat mengimpor hingga 10 item sekaligus. Jika Anda memiliki sejumlah besar item baru, kami sarankan Anda terlebih dahulu mengimpor data secara massal dan kemudian mengimpor data item satu per satu sesuai kebutuhan. Lihat Mengimpor data massal ke Amazon Personalisasi dengan pekerjaan impor kumpulan data.

Anda dapat menggunakan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau AWS SDKs untuk mengimpor item. Jika Anda mengimpor item yang `itemId` sama dengan item yang sudah ada di kumpulan data Item, Amazon Personalize menggantinya dengan item baru.

Untuk informasi tentang cara Amazon Personalisasi pembaruan memfilter untuk catatan baru dan bagaimana catatan baru memengaruhi rekomendasi, lihat. Mengimpor catatan individual ke dalam kumpulan data Amazon Personalize

Topik

- Mengimpor item satu per satu (konsol)
- Mengimpor item secara individual ()AWS CLI
- Mengimpor item secara individual ()AWS SDKs

Mengimpor item satu per satu (konsol)

Anda dapat mengimpor hingga 10 item ke kumpulan data Item sekaligus. Prosedur ini mengasumsikan bahwa Anda telah membuat dataset Item. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Untuk mengimpor item satu per satu (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Di halaman grup Dataset, pilih grup kumpulan data dengan kumpulan data Item yang ingin Anda impor itemnya.
3. Di panel navigasi, pilih Datasets.
4. Pada halaman Datasets, pilih dataset Item.
5. Di kanan atas halaman detail kumpulan data, pilih Ubah kumpulan data, lalu pilih Buat catatan.
6. Di Buat catatan item halaman, untuk Rekam input, masukkan detail item dalam format JSON. Nama dan nilai bidang item harus cocok dengan skema yang Anda gunakan saat membuat kumpulan data Item. Amazon Personalize menyediakan template JSON dengan nama bidang dan tipe data dari skema ini.
7. Pilih Buat catatan. Sebagai Respons, hasil impor terdaftar dan pesan sukses atau gagal ditampilkan.

Mengimpor item secara individual ()AWS CLI

Tambahkan satu atau beberapa item ke set data Item Anda menggunakan PutItems operasi. Anda dapat mengimpor hingga 10 item dengan satu `PutItems` panggilan. Bagian ini mengasumsikan bahwa Anda telah membuat dataset Item. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Gunakan `put-items` perintah berikut untuk menambahkan satu atau lebih item dengan AWS CLI. Ganti `dataset arn` dengan Nama Sumber Daya Amazon (ARN) dari kumpulan data Anda dan `item Id` dengan ID item. Jika item dengan item yang sama sudah `itemId` ada di kumpulan data Item Anda, Amazon Personalize menggantinya dengan yang baru.

Untuk`properties`, untuk setiap bidang dalam kumpulan data Item Anda, ganti `propertyName` dengan nama bidang dari skema Anda dalam kasus unta. Misalnya, GENRES akan menjadi `genres` dan CREATION_TIMESTAMP akan menjadi CreationTimeStamp. Ganti `item data` dengan data

untuk item tersebut. `CREATION_TIMESTAMP`data harus dalam format waktu epoch Unix dan dalam hitungan detik. Untuk data string kategoris, untuk menyertakan beberapa kategori untuk satu properti, pisahkan setiap kategori dengan pipe (`|`). Sebagai contoh, `\"Horror|Action\"`.

```
aws personalize-events put-items \
  --dataset-arn dataset arn \
  --items '[{
      "itemId": "item Id",
      "properties": "{\"propertyName\": "\item data\"}"
    },
    {
      "itemId": "item Id",
      "properties": "{\"propertyName\": "\item data\"}"
    }]'
```

Mengimpor item secara individual ()AWS SDKs

Tambahkan satu atau beberapa item ke set data Item Anda menggunakan PutItems operasi. Anda dapat mengimpor hingga 10 item dengan satu `PutItems` panggilan. Jika item dengan item yang sama sudah `itemId` ada di kumpulan data Item Anda, Amazon Personalize menggantinya dengan yang baru. Bagian ini mengasumsikan bahwa Anda telah membuat dataset Item. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Kode berikut menunjukkan cara menambahkan satu atau beberapa item ke kumpulan data Item Anda. Untuk setiap parameter nama properti, berikan nama bidang dari skema Anda dalam kasus unta. Misalnya, GENRES akan menjadi `genres` dan CREATION_TIMESTAMP akan menjadi. `creationTimestamp` Untuk setiap parameter nilai properti, berikan data untuk item tersebut. `CREATION_TIMESTAMP`data harus dalam format waktu epoch Unix dan dalam hitungan detik.

Untuk data string kategoris, untuk menyertakan beberapa kategori untuk satu properti, pisahkan setiap kategori dengan pipe (`|`). Sebagai contoh, `"Horror|Action"`.

SDK for Python (Boto3)

```
import boto3

personalize_events = boto3.client(service_name='personalize-events')

personalize_events.put_items(
    datasetArn = 'dataset arn',
    items = [{
```

```
      'itemId': 'item ID',
      'properties': "{\"propertyName\": \"item data\"}"
      },
      {
      'itemId': 'item ID',
      'properties': "{\"propertyName\": \"item data\"}"
      }]
)
```

SDK for Java 2.x

```
public static int putItems(PersonalizeEventsClient personalizeEventsClient,
                           String datasetArn,
                           String item1Id,
                           String item1PropertyName,
                           String item1PropertyValue,
                           String item2Id,
                           String item2PropertyName,
                           String item2PropertyValue) {

    int responseCode = 0;
    ArrayList<Item> items = new ArrayList<>();

    try {
        Item item1 = Item.builder()
                .itemId(item1Id)
                .properties(String.format("{\"%1$s\": \"%2$s\"}",
                        item1PropertyName, item1PropertyValue))
                .build();

        items.add(item1);

        Item item2 = Item.builder()
                .itemId(item2Id)
                .properties(String.format("{\"%1$s\": \"%2$s\"}",
                        item2PropertyName, item2PropertyValue))
                .build();

        items.add(item2);

        PutItemsRequest putItemsRequest = PutItemsRequest.builder()
                .datasetArn(datasetArn)
                .items(items)
```

```
                .build();

        responseCode =
 personalizeEventsClient.putItems(putItemsRequest).sdkHttpResponse().statusCode();
        System.out.println("Response code: " + responseCode);
        return responseCode;

    } catch (PersonalizeEventsException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return responseCode;
    }
```

SDK for JavaScript v3

```
import {
  PutItemsCommand,
  PersonalizeEventsClient,
} from "@aws-sdk/client-personalize-events";

const personalizeEventsClient = new PersonalizeEventsClient({
  region: "REGION",
});

// set the put items parameters
var putItemsParam = {
  datasetArn:
    "DATASET ARN",
  items: [
    {
      itemId: "itemId",
      properties: '{"column1Name": "value", "column2Name": "value"}',
    },
    {
      itemId: "itemId",
      properties: '{"column1Name": "value", "column2Name": "value"}',
    },
  ],
};
export const run = async () => {
  try {
    const response = await personalizeEventsClient.send(
      new PutItemsCommand(putItemsParam)
```

```
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Mengimpor tindakan secara individual

Setelah selesai Membuat skema dan dataset membuat kumpulan data Tindakan, Anda dapat mengimpor satu atau beberapa tindakan baru ke dalam kumpulan data secara individual. Saat Anda mengimpor tindakan secara individual, Anda akan tetap mengikuti set data Actions dengan impor batch kecil seiring pertumbuhan katalog. Anda dapat mengimpor hingga 10 tindakan sekaligus. Jika Anda memiliki sejumlah besar tindakan baru, kami sarankan Anda terlebih dahulu mengimpor data secara massal dan kemudian mengimpor data tindakan satu per satu sesuai kebutuhan. Lihat Mengimpor data massal ke Amazon Personalisasi dengan pekerjaan impor kumpulan data.

Anda dapat menggunakan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau AWS SDKs untuk mengimpor tindakan. Jika Anda mengimpor tindakan yang `actionId` sama dengan tindakan yang sudah ada di kumpulan data Tindakan, Amazon Personalize akan menggantinya dengan tindakan baru.

Untuk informasi tentang bagaimana catatan baru memengaruhi rekomendasi, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Topik

- Mengimpor tindakan secara individual (konsol)
- Mengimpor tindakan secara individual ()AWS CLI
- Mengimpor tindakan secara individual ()AWS SDKs

Mengimpor tindakan secara individual (konsol)

Anda dapat mengimpor hingga 10 tindakan ke dalam kumpulan data Tindakan sekaligus. Bagian ini mengasumsikan bahwa Anda telah membuat kumpulan data Actions. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Untuk mengimpor tindakan satu per satu (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data dengan kumpulan data Tindakan yang ingin Anda tambahkan.
3. Di panel navigasi, pilih Datasets.
4. Pada halaman Datasets, pilih dataset Actions.
5. Di kanan atas halaman detail kumpulan data, pilih Ubah kumpulan data, lalu pilih Buat catatan.
6. Di halaman Buat catatan tindakan, untuk Input Rekam, masukkan detail tindakan dalam format JSON. Nama dan nilai bidang tindakan harus sesuai dengan skema yang Anda gunakan saat membuat kumpulan data Tindakan. Amazon Personalize menyediakan template JSON dengan nama bidang dan tipe data dari skema ini.
7. Pilih Buat catatan (s). Sebagai Respons, hasil impor terdaftar dan pesan sukses atau gagal ditampilkan.

Mengimpor tindakan secara individual ()AWS CLI

Tambahkan satu atau beberapa tindakan ke kumpulan data Tindakan Anda menggunakan operasi `PutActions` API. Anda dapat mengimpor hingga 10 tindakan sekaligus. Bagian ini mengasumsikan bahwa Anda telah membuat kumpulan data Actions. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Gunakan `put-actions` perintah berikut untuk menambahkan satu atau beberapa tindakan dengan AWS CLI. Ganti `dataset arn` dengan Nama Sumber Daya Amazon (ARN) dari kumpulan data Anda dan `actionId` dengan ID tindakan. Jika tindakan yang sama sudah `actionId` ada di kumpulan data Tindakan Anda, Amazon Personalize menggantinya dengan yang baru.

Untuk`properties`, untuk setiap bidang dalam kumpulan data Tindakan Anda, ganti `propertyName` dengan nama bidang dari skema Anda dalam kasus unta. Misalnya, ACTION_EXPIRATION_TIMESTAMP akan menjadi dan CREATION_TIMESTAMP akan menjadi

CreationTimeStamp. `actionExpirationTimestamp` Ganti `property data` dengan data untuk properti.

```
aws personalize-events put-actions \
  --dataset-arn dataset arn \
  --actions '[{
      "actionId": "actionId",
      "properties": "{\"propertyName\": "\property data\"}"
    },
    {
      "actionId": "actionId",
      "properties": "{\"propertyName\": "\property data\"}"
    }]'
```

Mengimpor tindakan secara individual ()AWS SDKs

Tambahkan satu atau beberapa tindakan ke kumpulan data Tindakan Anda menggunakan PutActions operasi. Anda dapat mengimpor hingga 10 tindakan dengan satu `PutActions` panggilan. Jika tindakan yang sama sudah `actionId` ada di kumpulan data Tindakan Anda, Amazon Personalize menggantinya dengan yang baru. Bagian ini mengasumsikan bahwa Anda telah membuat kumpulan data Actions. Untuk informasi tentang membuat kumpulan data, lihat. Membuat skema dan dataset

Kode berikut menunjukkan cara menambahkan satu atau beberapa tindakan ke kumpulan data Tindakan Anda. Untuk setiap tindakan, tentukan`actionId`. Jika tindakan yang sama sudah `actionId` ada di kumpulan data Tindakan Anda, Amazon Personalize menggantinya dengan yang baru. Untuk`properties`, untuk setiap bidang tambahan dalam kumpulan data Tindakan Anda, ganti `propertyName` dengan nama bidang dari skema Anda dalam kasus unta. Misalnya, ACTION_EXPIRATION_TIMESTAMP akan menjadi dan CREATION_TIMESTAMP akan menjadi CreationTimeStamp. `actionExpirationTimestamp` Ganti `property data` dengan data untuk properti.

```
import boto3

personalize_events = boto3.client(service_name='personalize-events')

personalize_events.put_actions(
    datasetArn = 'dataset arn',
    actions = [{
      'actionId': 'actionId',
```

```
        'properties': "{\"propertyName\": \"property value\"}"
        },
        {
        'actionId': 'actionId',
        'properties': "{\"propertyName\": \"property value\"}"
        }]
)
```

Menganalisis kualitas dan kuantitas data di Amazon Personalisasi kumpulan data

Setelah mengimpor data ke dalam kumpulan data interaksi Item, Pengguna, atau Item, Anda dapat menggunakan konsol Amazon Personalize untuk menganalisis data. Anda dapat mempelajari data Anda melalui wawasan data dan statistik kolom dan baris. Dan Anda dapat mempelajari tindakan apa yang dapat Anda ambil untuk meningkatkan data Anda. Tindakan ini dapat membantu Anda memenuhi persyaratan sumber daya Amazon Personalize, seperti persyaratan pelatihan model, atau dapat mengarah pada rekomendasi yang lebih baik.

> **Important**
>
> Anda tidak dapat menggunakan konsol Amazon Personalize untuk menganalisis data dalam kumpulan data Interaksi atau Tindakan.

Setelah Anda membuat perubahan yang disarankan, Anda dapat mengimpor data Anda lagi dan melihat apakah Anda menyelesaikan masalah atau meningkatkan statistik kumpulan data. Untuk informasi tentang memperbarui data, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Jika Anda tidak melihat wawasan apa pun, data Anda selaras dengan ekspektasi data Amazon Personalisasi. Anda dapat menganalisis data dalam grup kumpulan data Domain atau grup kumpulan data kustom.

Saat menghasilkan wawasan dan menghitung statistik, Amazon Personalize mempertimbangkan semua data massal dan streaming dari pengguna non-anonim. Peristiwa dari pengguna anonim tidak dipertimbangkan sampai Anda mengaitkannya dengan file`userId`. Untuk informasi selengkapnya, lihat Merekam acara untuk pengguna anonim.

Topik

- Izin yang diperlukan untuk menganalisis data
- Wawasan data
- Melihat wawasan dan statistik kumpulan data

Izin yang diperlukan untuk menganalisis data

Jika Anda memberi pengguna akses penuh ke Amazon Personalize, tidak diperlukan perubahan izin. Jika Anda hanya memberi pengguna izin yang diperlukan untuk melakukan tugas di Amazon Personalize, kebijakan (IAM) AWS Identity and Access Management Anda harus menyertakan tindakan wawasan data tambahan berikut.

- personalisasi: CreateDataInsightsJob
- personalisasi: ListDataInsightsJobs
- personalisasi: DescribeDataInsightsJob
- personalisasi: GetDataInsights

Wawasan data

Berikut ini adalah kemungkinan wawasan data yang dapat Anda hasilkan di Amazon Personalisasi.

Wawasan	Tindakan	Dataset terkait
Dataset Interaksi hanya memiliki interaksi X. Pelatihan model membutuhkan minimal 1.000 interaksi. Kami merekomen dasikan setidaknya 50.000.	Impor Y catatan interaksi unik tambahan sebelum melatih model.	Interaksi item
Dataset Interaksi hanya memiliki X pengguna unik dengan dua atau lebih interaksi. Pelatihan model membutuhkan setidaknya 25 pengguna tersebut. Kami merekomendasikan setidaknya 1.000.	Impor setidaknya 2 catatan interaksi masing-masing untuk pengguna tambahan Y.	Interaksi item
X% item dalam kumpulan data Item tidak memiliki interaksi dalam kumpulan data Interaksi, jadi mungkin tidak direkomen dasikan.	Pastikan Anda mengimpor semua data interaksi dan memeriksa ketidakcocokan IDs antara item dan kumpulan data interaksi Anda. Periksa Statistik Dataset di bawah ini untuk item dan kumpulan data interaksi Anda untuk memastikan Anda telah mengimpor jumlah baris yang	Interaksi item dan Item

Wawasan	Tindakan	Dataset terkait
	diharapkan. Jika kasus penggunaan atau resep Anda menggunakan eksploras i, ubah konfigurasi eksplorasi untuk merekomendasikan lebih banyak item tanpa data interaksi.	
10% pengguna dalam kumpulan data Pengguna tidak memiliki interaksi dalam kumpulan data Interaksi. Pengguna ini akan menerima rekomendasi untuk item populer.	Pastikan Anda mengimpor semua data interaksi dan memeriksa ketidakcocokan IDs antara pengguna dan kumpulan data interaksi. Periksa Statistik Dataset di bawah ini untuk pengguna dan kumpulan data interaksi Anda untuk memastikan Anda telah mengimpor jumlah baris yang diharapkan. Impor interaksi tambahan apa pun sehingga lebih banyak pengguna memiliki data interaksi.	Interaksi item dan Pengguna
<Users or Items or Interactions>Datas et memiliki baris X% dengan nilai yang hilang. Ini dapat berdampak negatif pada rekomendasi. Kami menyarankan agar semua bidang wajib dan opsional setidaknya 70% persen selesai.	Impor catatan lengkap tambahan, atau impor data lagi tanpa baris yang tidak lengkap, atau impor data lagi dengan nilai yang hilang diganti dengan data pengganti, seperti rata-rata untuk kolom numerik atau nilai paling umum untuk kolom kategoris.	Setiap
Kolom berikut dalam <datasetT ype>kumpulan data kurang dari 70% lengkap: <ColumnName, ColumnName... >. Jika data ini termasuk dalam pelatihan , itu dapat berdampak negatif pada rekomendasi. Kami merekomendasikan bahwa kolom yang memungkinkan nilai nol setidaknya 70% selesai.	Impor catatan lengkap tambahan, atau impor data lagi tanpa baris yang tidak lengkap, atau impor data lagi dengan nilai yang hilang diganti dengan data pengganti, seperti rata-rata untuk kolom numerik atau nilai paling umum untuk kolom kategoris.	Setiap

Wawasan	Tindakan	Dataset terkait
Kolom (numerik) berikut memiliki outlier: <ColumnName, ColumnName... >. Outlier tidak selalu menjadi masalah, tetapi terkadang berdampak negatif pada rekomendasi.	Menggunakan Statistik Kolom di bawah ini, periksa apakah nilai min dan maks untuk kolom ini sesuai dengan harapan Anda. Jika nilai ini tidak terduga, periksa data di kolom ini untuk ketidakakuratan dan tinjau pengumpulan data dan pemrosesan data Anda untuk masalah.	Setiap
Kolom berikut memiliki lebih dari 1000 kategori yang mungkin: <ColumnName, ColumnName... >. Jika data ini termasuk dalam pelatihan, itu dapat berdampak negatif pada rekomendasi: <ColumnNa me, ColumnName... >.	Periksa data kategoris Anda untuk masalah, seperti kategori duplikat yang disebabkan oleh variasi ejaan. Selesaikan ketidakakuratan dan impor data lagi.	Setiap
Kolom metadata tekstual berikut kurang dari 85% persen lengkap dan tidak akan digunakan dalam pelatihan model: <ColumnName, ColumnName... >.	Impor baris tambahan atau impor baris lagi dengan data teks untuk kolom ini.	Item
Dataset Interaksi memiliki lebih dari 10 jenis peristiwa unik, yang akan menyebabkan pelatihan model gagal.	Periksa kolom jenis acara Anda untuk ketidakakuratan seperti jenis peristiwa duplikat yang disebabkan oleh variasi ejaan. Hapus jenis acara yang tidak perlu dan impor data lagi.	Interaksi item
Dataset Interaksi memiliki stempel waktu yang sama untuk semua catatan. Jika Anda menggunakan resep USER_SEGM ENTATION dan semua catatan memiliki stempel waktu yang sama, pelatihan model akan gagal.	Periksa data Anda untuk masalah stempel waktu dan ganti stempel waktu duplikat dengan stempel waktu unik.	Interaksi item

Melihat wawasan dan statistik kumpulan data

Untuk melihat wawasan dan statistik tentang data Anda di Amazon Personalisasi kumpulan data, navigasikan ke kumpulan data Anda di konsol Amazon Personalize dan pilih jalankan analisis.

> **Note**
>
> Anda harus terlebih dahulu mengimpor data sebelum Amazon Personalize dapat menganalisisnya. Untuk informasi selengkapnya, lihat Mengimpor data pelatihan ke Amazon Personalize dataset

Untuk melihat wawasan dan statistik

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
3. Dari panel navigasi, di bawah Datasets pilih Analisis data.
4. Di kanan atas, pilih Jalankan analisis. Amazon Personalize mulai menganalisis data Anda. Ini bisa memakan waktu hingga 15 menit. Jika berhasil, hasilnya muncul di halaman ini.
5. Di Wawasan, gunakan yang berikut ini untuk memfilter wawasan yang muncul.
   - Untuk menemukan wawasan yang menyertakan bahasa tertentu, masukkan kriteria Anda di Temukan wawasan. Saat Anda memasukkan teks, daftar akan diperbarui untuk hanya menyertakan wawasan dengan string yang tepat dalam wawasan atau tindakan yang disarankan.
   - Untuk memfilter wawasan berdasarkan jenis kumpulan data, ubah Semua kumpulan data ke jenis kumpulan data tertentu. Daftar diperbarui untuk menyertakan hanya wawasan yang terkait dengan kumpulan data ini.
6. Untuk melihat statistik kumpulan data untuk kumpulan data, lakukan hal berikut.
   - Untuk melihat detail umum dan statistik tentang kumpulan data, seperti jumlah baris, pengguna unik, dan item unik dalam kumpulan data Interaksi, perluas bagian untuk kumpulan data.
   - Untuk melihat statistik terperinci untuk kolom, perluas bagian kumpulan data, pilih Statistik tingkat kolom dan pilih tombol radio untuk kolom.

7. Perbaiki masalah apa pun dalam data Anda, impor lagi, dan jalankan analisis lain untuk memverifikasi. Untuk informasi selengkapnya tentang mengimpor data lagi, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Rekomendasi domain di Amazon Personalisasi

Jika Anda membuat grup kumpulan data Domain, setelah mengimpor data, Anda siap membuat rekomendasi untuk kasus penggunaan domain Anda. Rekomendasi adalah sumber daya grup kumpulan data Domain yang menghasilkan rekomendasi. Anda menggunakan pemberi rekomendasi dalam aplikasi Anda untuk mendapatkan rekomendasi waktu nyata dengan GetRecommendations operasi.

Saat membuat pemberi rekomendasi, Anda menentukan kasus penggunaan dan Amazon Personalize melatih model yang mendukung pemberi rekomendasi dengan konfigurasi terbaik untuk kasus penggunaan. Setiap kasus penggunaan memiliki persyaratan API yang berbeda untuk mendapatkan rekomendasi. Untuk daftar kasus penggunaan pemberi rekomendasi berdasarkan domain, lihatMemilih kasus penggunaan. Anda dapat membuat paling banyak 15 rekomendasi per wilayah.

Amazon Personalize secara otomatis melatih ulang model yang mendukung rekomendasi Anda setiap 7 hari. Ini adalah pelatihan ulang penuh yang membuat model yang sama sekali baru berdasarkan keseluruhan data dalam kumpulan data Anda. Dengan Pilihan teratas untuk Anda dan Direkomendasikan untuk kasus penggunaan Anda, Amazon Personalize memperbarui model yang ada setiap dua jam untuk menyertakan item baru dalam rekomendasi dengan eksplorasi.

Saat Anda membuat pemberi rekomendasi, Anda dapat melakukan hal berikut:

- Aktifkan metadata item dalam rekomendasi. Untuk informasi selengkapnya, lihat Mengaktifkan metadata dalam rekomendasi.
- Konfigurasikan kolom yang digunakan saat melatih model yang mendukung pemberi rekomendasi Anda. Untuk informasi selengkapnya, lihat Mengonfigurasi kolom yang digunakan saat membuat rekomendasi domain Amazon Personalize.
- Untuk Pilihan teratas untuk kasus penggunaan Anda atau Direkomendasikan untuk Anda, Anda dapat mengonfigurasi eksplorasi. Untuk informasi selengkapnya, lihat Mengkonfigurasi eksplorasi untuk pemberi rekomendasi domain.

Setelah Anda membuat pemberi rekomendasi, Anda dapat melakukan hal berikut:

- Evaluasi pemberi rekomendasi — Anda dapat mengevaluasi kinerja pemberi rekomendasi Anda melalui metrik offline dan online. Untuk informasi selengkapnya, lihat Mengevaluasi rekomendasi domain Personalisasi Amazon.

- Hentikan dan mulai ulang pemberi rekomendasi — Jika Anda ingin menjeda penagihan untuk pemberi rekomendasi aktif, Anda dapat menghentikan pemberi rekomendasi dan memulai ulang nanti. Untuk informasi selengkapnya, lihat Menghentikan pemberi rekomendasi.
- Perbarui konfigurasi pemberi rekomendasi — Anda dapat memperbarui kolom yang digunakan pemberi rekomendasi dalam pelatihan dan memperbarui kapasitas permintaan pemberi rekomendasi. Untuk informasi selengkapnya, lihat Memperbarui pemberi rekomendasi.
- Hapus pemberi rekomendasi — Anda dapat menghapus pemberi rekomendasi dengan operasi. DeleteRecommender Atau Anda dapat menghapus pemberi rekomendasi dari halaman detail pemberi rekomendasi di konsol Amazon Personalize.

Topik

- Status rekomendasi
- Permintaan rekomendasi minimum per detik dan auto-scaling
- Membuat rekomendasi domain di Amazon Personalize
- Mengaktifkan metadata dalam rekomendasi untuk pemberi rekomendasi domain di Amazon Personalize
- Mengonfigurasi kolom yang digunakan saat membuat rekomendasi domain Amazon Personalize
- Mengkonfigurasi eksplorasi untuk pemberi rekomendasi domain
- Mengevaluasi rekomendasi domain Personalisasi Amazon
- Memperbarui pemberi rekomendasi
- Menghentikan pemberi rekomendasi

Status rekomendasi

Seorang pemberi rekomendasi dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Untuk mendapatkan status pemberi rekomendasi, buka halaman Rekomendasi di konsol Amazon Personalize atau gunakan operasi. DescribeRecommender

Permintaan rekomendasi minimum per detik dan auto-scaling

> **Important**
>
> Tinggi `minRecommendationRequestsPerSecond` akan meningkatkan tagihan Anda. Sebaiknya mulai dengan 1 untuk `minRecommendationRequestsPerSecond` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minRecommendationRequestsPerSecond` seperlunya.

Saat membuat pemberi rekomendasi, Anda dapat mengonfigurasi permintaan rekomendasi minimum pemberi rekomendasi per detik. Permintaan rekomendasi minimum per detik (`minRecommendationRequestsPerSecond`) menentukan throughput permintaan rekomendasi dasar yang disediakan oleh Amazon Personalize. minRecommendationRequestsPerSecond Defaultnya adalah1. Permintaan rekomendasi adalah `GetRecommendations` operasi tunggal. Throughput permintaan diukur dalam permintaan per detik dan Amazon Personalize menggunakan permintaan Anda per detik untuk mendapatkan permintaan Anda per jam dan harga penggunaan pemberi rekomendasi Anda.

Jika permintaan Anda per detik meningkat lebih dari ituminRecommendationRequestsPerSecond, Amazon Personalize secara otomatis menskalakan kapasitas yang disediakan ke atas dan ke bawah, tetapi tidak pernah di bawah ini. `minRecommendationRequestsPerSecond` Ada penundaan waktu singkat sementara kapasitas meningkat yang dapat menyebabkan hilangnya permintaan.

Tagihan Anda lebih besar dari permintaan minimum per jam (berdasarkan minRecommendationRequestsPerSecond) atau jumlah permintaan yang sebenarnya. Throughput permintaan aktual yang digunakan dihitung sebagai rata-rata requests/second dalam jendela satu jam. Sebaiknya mulai dengan defaultminRecommendationRequestsPerSecond, lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan kemudian tingkatkan `minRecommendationRequestsPerSecond` seperlunya.

Membuat rekomendasi domain di Amazon Personalize

Anda dapat membuat pemberi rekomendasi dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau. AWS SDKs Berikut ini mencakup langkah-langkah terperinci untuk membuat pemberi rekomendasi dengan konsol Amazon Personalize dan contoh kode yang menunjukkan cara membuat pemberi rekomendasi hanya dengan bidang yang diperlukan.

- Untuk contoh kode yang menunjukkan cara mengaktifkan metadata dalam rekomendasi, lihat. Mengaktifkan metadata dalam rekomendasi
- Untuk contoh kode yang menunjukkan cara mengonfigurasi kolom yang digunakan saat melatih model yang mendukung pemberi rekomendasi Anda, lihat. Mengonfigurasi kolom yang digunakan saat membuat rekomendasi domain Amazon Personalize
- Untuk contoh kode yang menunjukkan cara mengonfigurasi eksplorasi untuk kasus `Recommended for you` penggunaan `Top picks for your` atau, lihatMengkonfigurasi eksplorasi untuk pemberi rekomendasi domain.

Topik

- Membuat rekomendasi (konsol)
- Membuat pemberi rekomendasi ()AWS CLI
- Membuat pemberi rekomendasi ()AWS SDKs

Membuat rekomendasi (konsol)

> **Important**
>
> Tinggi `minRecommendationRequestsPerSecond` akan meningkatkan tagihan Anda. Sebaiknya mulai dengan 1 untuk `minRecommendationRequestsPerSecond` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minRecommendationRequestsPerSecond` seperlunya. Untuk informasi selengkapnya, lihat Permintaan rekomendasi minimum per detik dan auto-scaling.

Buat rekomendasi untuk setiap kasus penggunaan Anda dengan konsol Amazon Personalize sebagai berikut. Jika Anda baru saja membuat grup dataset Domain Anda dan Anda sudah berada di halaman Ikhtisar, lewati ke langkah 3.

Untuk membuat pemberi rekomendasi

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Domain Anda.

3. Pada Langkah 3, pilih Gunakan pemberi rekomendasi <domain name>dan pilih Buat pemberi rekomendasi.
4. Pada halaman Pilih kasus penggunaan, pilih kasus penggunaan yang ingin Anda buat pemberi rekomendasi dan berikan masing-masing nama Rekomendasi. Amazon Personalize membuat pemberi rekomendasi untuk setiap kasus penggunaan yang Anda pilih. Kasus penggunaan yang tersedia tergantung pada domain Anda. Untuk informasi tentang memilih kasus penggunaan, lihatMemilih kasus penggunaan.
5. Pilih Berikutnya.
6. Pada halaman Konfigurasi lanjutan, konfigurasikan setiap pemberi rekomendasi tergantung pada kebutuhan bisnis Anda:

   - Untuk setiap kumpulan data yang digunakan oleh kasus penggunaan pemberi rekomendasi, Anda dapat memilih kolom yang dipertimbangkan Amazon Personalize saat melatih model yang mendukung pemberi rekomendasi Anda. Secara default, Amazon Personalize menggunakan semua kolom yang dapat digunakan saat pelatihan. Untuk informasi selengkapnya, lihat Mengonfigurasi kolom yang digunakan saat membuat rekomendasi domain Amazon Personalize.
   - Anda dapat memodifikasi Permintaan rekomendasi minimum per detik untuk menentukan kapasitas permintaan minimum baru untuk pemberi rekomendasi Anda. Tinggi `minRecommendationRequestsPerSecond` akan meningkatkan tagihan Anda. Sebaiknya mulai dengan 1 (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minRecommendationRequestsPerSecond` seperlunya. Untuk mengetahui informasi selengkapnya, lihat Permintaan rekomendasi minimum per detik dan auto-scaling.
   - Jika Anda ingin kemampuan untuk menyertakan metadata kumpulan data Item dengan rekomendasi, pilih Kembalikan metadata item dalam hasil rekomendasi. Jika diaktifkan, Anda dapat menentukan kolom dari kumpulan data Item dalam permintaan rekomendasi atau peringkat yang dipersonalisasi. Amazon Personalize mengembalikan data ini untuk setiap item dalam respons rekomendasi.

     Untuk mengaktifkan metadata, Anda harus memiliki kumpulan data Item dengan kolom metadata.
   - Untuk `Top picks for your` atau kasus `Recommended for you` penggunaan, secara opsional membuat perubahan pada konfigurasi eksplorasi. Eksplorasi melibatkan pengujian rekomendasi item yang berbeda untuk mempelajari bagaimana pengguna merespons item

dengan data interaksi yang sangat sedikit. Gunakan bidang berikut untuk mengonfigurasi eksplorasi:

- Penekanan pada penjelajahan item yang kurang relevan (bobot eksplorasi) - Konfigurasikan berapa banyak yang harus dijelajahi. Tentukan nilai desimal antara 0 hingga 1. Defaultnya adalah 0,3. Semakin dekat nilainya ke 1, semakin banyak eksplorasi. Dengan lebih banyak eksplorasi, rekomendasi mencakup lebih banyak item dengan lebih sedikit data interaksi item atau relevansi berdasarkan perilaku sebelumnya. Pada nol, tidak ada eksplorasi terjadi dan rekomendasi didasarkan pada data saat ini (relevansi).
- Batas usia item eksplorasi - Tentukan usia item maksimum dalam beberapa hari sejak interaksi terbaru di semua item dalam kumpulan data interaksi item. Ini mendefinisikan ruang lingkup eksplorasi item berdasarkan usia item. Amazon Personalize menentukan usia item berdasarkan stempel waktu pembuatannya atau, jika data stempel waktu pembuatan tidak ada, data interaksi item. Untuk informasi selengkapnya cara Amazon Personalize menentukan usia item, lihat. Data stempel waktu pembuatan

  Untuk meningkatkan item yang dipertimbangkan Amazon Personalize selama eksplorasi, masukkan nilai yang lebih besar. Minimal adalah 1 hari dan defaultnya adalah 30 hari. Rekomendasi mungkin mencakup item yang lebih tua dari potongan usia item yang Anda tentukan. Ini karena item ini relevan dengan pengguna dan eksplorasi tidak mengidentifikasinya.
- Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya

7. Untuk membuat rekomendasi untuk setiap kasus penggunaan Anda, pilih Buat pemberi rekomendasi.

   Anda dapat memantau status masing-masing pemberi rekomendasi di halaman Rekomendasi. Ketika status pemberi rekomendasi Anda Aktif, Anda dapat menggunakannya dalam aplikasi Anda untuk mendapatkan rekomendasi.

Membuat pemberi rekomendasi ()AWS CLI

Gunakan AWS CLI kode berikut untuk membuat pemberi rekomendasi untuk kasus penggunaan domain. Jalankan kode ini untuk setiap kasus penggunaan domain Anda. Untuk`recipeArn`, berikan Nama Sumber Daya Amazon (ARN) untuk kasus penggunaan Anda. Kasus penggunaan

yang tersedia tergantung pada domain Anda. Untuk daftar kasus penggunaan dan ARNs lihat merekaMemilih kasus penggunaan.

```
aws personalize create-recommender \
--name recommender name \
--dataset-group-arn dataset group ARN \
--recipe-arn recipe ARN
```

Membuat pemberi rekomendasi ()AWS SDKs

Buat pemberi rekomendasi untuk kasus penggunaan domain dengan kode berikut. Beri nama pada pemberi rekomendasi Anda dan berikan Nama Sumber Daya Amazon (ARN) grup kumpulan data Domain Anda. Untuk`recipeArn`, berikan ARN untuk kasus penggunaan Anda. Jalankan kode ini untuk setiap kasus penggunaan domain Anda. Kasus penggunaan yang tersedia tergantung pada domain Anda. Untuk daftar kasus penggunaan, persyaratannya ARNs, dan persyaratannya, lihatMemilih kasus penggunaan.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

create_recommender_response = personalize.create_recommender(
  name = 'recommender name',
  recipeArn = 'recipe ARN',
  datasetGroupArn = 'dataset group ARN'
)

recommender_arn = create_recommender_response['recommenderArn']

print('Recommender ARN:' + recommender_arn)
```

SDK for Java 2.x

```
    public static String createRecommender(PersonalizeClient personalizeClient,
            String name,
            String datasetGroupArn,
            String recipeArn) {

        long maxTime = 0;
```

```
        long waitInMilliseconds = 30 * 1000; // 30 seconds
        String recommenderStatus = "";

        try {
            CreateRecommenderRequest createRecommenderRequest =
CreateRecommenderRequest.builder()
                    .datasetGroupArn(datasetGroupArn)
                    .name(name)
                    .recipeArn(recipeArn)
                    .build();

            CreateRecommenderResponse recommenderResponse = personalizeClient
                    .createRecommender(createRecommenderRequest);
            String recommenderArn = recommenderResponse.recommenderArn();
            System.out.println("The recommender ARN is " + recommenderArn);

            DescribeRecommenderRequest describeRecommenderRequest =
DescribeRecommenderRequest.builder()
                    .recommenderArn(recommenderArn)
                    .build();

            maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            while (Instant.now().getEpochSecond() < maxTime) {

                recommenderStatus =
personalizeClient.describeRecommender(describeRecommenderRequest).recommender()
                        .status();
                System.out.println("Recommender status: " + recommenderStatus);

                if (recommenderStatus.equals("ACTIVE") ||
recommenderStatus.equals("CREATE FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }
            return recommenderArn;

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
```

```
            System.exit(1);
        }
        return "";
    }
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { CreateRecommenderCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

// set the recommender's parameters
export const createRecommenderParam = {
  name: "RECOMMENDER_NAME",                    /* required */
  recipeArn: "RECIPE_ARN",                     /* required */
  datasetGroupArn: "DATASET_GROUP_ARN"         /* required */
}

export const run = async () => {
  try {
    const response = await personalizeClient.send(new
 CreateRecommenderCommand(createRecommenderParam));
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Mengaktifkan metadata dalam rekomendasi untuk pemberi rekomendasi domain di Amazon Personalize

> **Important**
>
> Ketika Anda mengaktifkan metadata dalam rekomendasi, Anda dikenakan biaya tambahan. Untuk informasi selengkapnya, lihat Harga Amazon Personalisasi.

Saat membuat pemberi rekomendasi, Anda dapat mengaktifkan opsi untuk menyertakan metadata item dari kumpulan data Item Anda dengan hasil rekomendasi. Jika diaktifkan, Anda dapat menentukan kolom dari kumpulan data Item dalam permintaan rekomendasi. Amazon Personalize mengembalikan data ini untuk setiap item dalam respons rekomendasi.

Anda dapat menggunakan metadata untuk memperkaya rekomendasi di antarmuka pengguna, seperti menambahkan genre film ke komidi putar. Atau Anda dapat menggunakannya untuk menilai kualitas rekomendasi secara visual. Jika Anda menggunakan AI generatif di aplikasi, Anda dapat menyambungkan metadata ke prompt AI untuk menghasilkan konten yang lebih relevan. Untuk informasi selengkapnya tentang menggunakan Amazon Personalize dengan AI generatif, lihat. Amazon Personalisasi dan AI generatif

Untuk menambahkan metadata ke rekomendasi, Anda harus memiliki kumpulan data Item dengan kolom metadata. Anda tidak harus menggunakan metadata dalam pelatihan. Untuk informasi tentang membuat kumpulan data, lihatMembuat skema dan dataset. Untuk informasi yang memperbarui data dalam kumpulan data, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Contoh kode berikut menunjukkan cara mengaktifkan opsi untuk menyertakan metadata item dengan AWS CLI atau. AWS SDKs Untuk melakukannya dengan konsol Amazon Personalize, Anda mengaktifkan metadata di halaman Konfigurasi lanjutan saat membuat pemberi rekomendasi. Untuk informasi selengkapnya, lihat Membuat rekomendasi (konsol).

Mengaktifkan metadata ()AWS CLI

Jika Anda memiliki kumpulan data Item dan ingin opsi menyertakan metadata saat Anda mendapatkan rekomendasi, atur `enableMetadataWithRecommendations` ke `true` dalam. `recommender-config`

```
aws personalize create-recommender \
```

```
--name recommender name \
--dataset-group-arn dataset group \
--recipe-arn recipe ARN \
--recommender-config "{\"enableMetadataWithRecommendations\": "true"}"
```

Mengaktifkan metadata (SDKS)AWS

Jika Anda memiliki kumpulan data Item dan ingin opsi menyertakan metadata saat Anda mendapatkan rekomendasi, atur `enableMetadataWithRecommendations` ke `true` dalam. `recommender-config`

```
import boto3

personalize = boto3.client('personalize')

create_recommender_response = personalize.create_recommender(
  name = 'recommender name',
  recipeArn = 'recipe name',
  datasetGroupArn = 'dataset group ARN',
  recommenderConfig = {"enableMetadataWithRecommendations": True}
)

recommender_arn = create_recommender_response['recommenderArn']

print('Recommender ARN:' + recommender_arn)
```

Mengonfigurasi kolom yang digunakan saat membuat rekomendasi domain Amazon Personalize

Saat membuat pemberi rekomendasi, Anda dapat memodifikasi kolom yang dipertimbangkan Amazon Personalize saat melatih model yang mendukung pemberi rekomendasi Anda.

Anda dapat melakukan ini untuk bereksperimen dengan kombinasi data pelatihan yang berbeda. Atau Anda mungkin mengecualikan kolom tanpa data yang berarti. Misalnya, Anda mungkin memiliki kolom yang ingin Anda gunakan hanya untuk memfilter rekomendasi. Anda dapat mengecualikan kolom ini dari pelatihan dan Amazon Personalize mempertimbangkannya hanya saat memfilter.

Anda tidak dapat mengecualikan kolom EVENT_TYPE. Secara default, Amazon Personalize menggunakan semua kolom yang dapat digunakan saat pelatihan. Data berikut selalu dikecualikan dari pelatihan:

- Kolom dengan tipe data boolean
- Data tayangan
- Bidang string kustom yang tidak kategoris atau tekstual

Anda tidak dapat menyertakan data tayangan dalam pelatihan, tetapi jika kasus penggunaan atau resep Anda menggunakannya, Amazon Personalize menggunakan data tayangan untuk memandu eksplorasi saat Anda mendapatkan rekomendasi.

Contoh kode berikut menunjukkan bagaimana mengkonfigurasi kolom yang digunakan saat pelatihan dengan AWS CLI atau AWS SDKs. Untuk melakukannya dengan konsol Amazon Personalize, Anda menentukan kolom yang akan digunakan pada halaman Konfigurasi lanjutan saat Anda membuat pemberi rekomendasi. Untuk informasi selengkapnya, lihat Membuat rekomendasi (konsol).

Mengkonfigurasi kolom yang digunakan saat pelatihan ()AWS CLI

Untuk mengecualikan kolom dari pelatihan, berikan `excludedDatasetColumns` objek di `trainingDataConfig` sebagai bagian dari konfigurasi pemberi rekomendasi. Untuk setiap kunci dalam objek, berikan tipe dataset. Untuk setiap nilai, berikan daftar kolom untuk dikecualikan. Untuk informasi selengkapnya, lihat Mengonfigurasi kolom yang digunakan saat membuat rekomendasi domain Amazon Personalize.

```
aws personalize create-recommender \
--name recommender name \
--dataset-group-arn dataset group ARN \
--recipe-arn recipe ARN \
--recommender-config "{\"trainingDataConfig\": {\"excludedDatasetColumns\":
 { \"datasetType\" : [ \"column1Name\", \"column2Name\"]}}}"
```

Mengkonfigurasi kolom yang digunakan saat pelatihan ()AWS SDKs

Untuk mengecualikan kolom dari pelatihan, berikan `excludedDatasetColumns` objek di `trainingDataConfig` sebagai bagian dari konfigurasi pemberi rekomendasi. Untuk setiap kunci, berikan jenis dataset. Untuk setiap nilai, berikan daftar kolom untuk dikecualikan. Kode berikut menunjukkan cara mengecualikan kolom dari pelatihan saat Anda membuat pemberi rekomendasi. Untuk informasi selengkapnya, lihat Mengonfigurasi kolom yang digunakan saat membuat rekomendasi domain Amazon Personalize.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

create_recommender_response = personalize.create_recommender(
  name = 'recommender name',
  recipeArn = 'recipe name',
  datasetGroupArn = 'dataset group ARN',
  recommenderConfig = {
    "trainingDataConfig": {
      "excludedDatasetColumns": {
        "datasetType": ["COLUMN_A", "COLUMN_B"]
      }
    }
  }
)

recommender_arn = create_recommender_response['recommenderArn']

print('Recommender ARN:' + recommender_arn)
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { CreateRecommenderCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

// set the recommender's parameters
export const createRecommenderParam = {
  name: "RECOMMENDER_NAME",             /* required */
  recipeArn: "RECIPE_ARN",              /* required */
  datasetGroupArn: "DATASET_GROUP_ARN", /* required */
  recommenderConfig: {
    trainingDataConfig: {
      excludedDatasetColumns: {
        "DATASET_TYPE": ["COLUMN_A", "COLUMN_B"]
```

```
      }
    }
  }
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(new
 CreateRecommenderCommand(createRecommenderParam));
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Mengkonfigurasi eksplorasi untuk pemberi rekomendasi domain

Untuk `Top picks for your` atau kasus `Recommended for you` penggunaan, Amazon Personalize menggunakan eksplorasi saat merekomendasikan item. Eksplorasi melibatkan pengujian rekomendasi item yang berbeda untuk mempelajari bagaimana pengguna merespons item dengan data interaksi yang sangat sedikit. Anda dapat mengonfigurasi eksplorasi dengan yang berikut ini:

- Penekanan pada penjelajahan item yang kurang relevan (bobot eksplorasi) - Konfigurasikan berapa banyak yang harus dijelajahi. Tentukan nilai desimal antara 0 hingga 1. Defaultnya adalah 0,3. Semakin dekat nilainya ke 1, semakin banyak eksplorasi. Dengan lebih banyak eksplorasi, rekomendasi mencakup lebih banyak item dengan lebih sedikit data interaksi item atau relevansi berdasarkan perilaku sebelumnya. Pada nol, tidak ada eksplorasi terjadi dan rekomendasi didasarkan pada data saat ini (relevansi).
- Batas usia item eksplorasi - Tentukan usia item maksimum dalam beberapa hari sejak interaksi terbaru di semua item dalam kumpulan data interaksi item. Ini mendefinisikan ruang lingkup eksplorasi item berdasarkan usia item. Amazon Personalize menentukan usia item berdasarkan stempel waktu pembuatannya atau, jika data stempel waktu pembuatan tidak ada, data interaksi item. Untuk informasi selengkapnya cara Amazon Personalize menentukan usia item, lihat. Data stempel waktu pembuatan

  Untuk meningkatkan item yang dipertimbangkan Amazon Personalize selama eksplorasi, masukkan nilai yang lebih besar. Minimal adalah 1 hari dan defaultnya adalah 30 hari.

Rekomendasi mungkin mencakup item yang lebih tua dari potongan usia item yang Anda tentukan. Ini karena item ini relevan dengan pengguna dan eksplorasi tidak mengidentifikasinya.

Contoh kode berikut menunjukkan cara mengonfigurasi eksplorasi untuk pemberi rekomendasi dengan AWS CLI atau file. AWS SDKs Untuk melakukannya dengan konsol Amazon Personalize, Anda menentukan konfigurasi eksplorasi pada halaman Konfigurasi lanjutan saat membuat pemberi rekomendasi. Untuk informasi selengkapnya, lihat Membuat rekomendasi (konsol).

Mengkonfigurasi eksplorasi ()AWS CLI

Kode berikut menunjukkan cara mengonfigurasi eksplorasi saat Anda membuat pemberi rekomendasi untuk kasus `Top picks for you` penggunaan. Contoh menggunakan nilai default.

Jika Anda memiliki kumpulan data Item dan ingin opsi menyertakan metadata saat Anda mendapatkan rekomendasi, perbarui `enableMetadataWithRecommendations` bidang `recommender-config` untuk menambahkan dan menyetelnya. `true`

```
aws personalize create-recommender \
--name recommender name \
--dataset-group-arn dataset group ARN \
--recipe-arn arn:aws:personalize:::recipe/aws-vod-top-picks \
--recommender-config "{\"itemExplorationConfig\":{\"explorationWeight\":\"0.3\",
\"explorationItemAgeCutOff\":\"30\"}}"
```

Mengkonfigurasi eksplorasi ()AWS SDKs

Untuk `Top picks for your` atau kasus `Recommended for you` penggunaan, Amazon Personalize menggunakan eksplorasi saat merekomendasikan item. Eksplorasi melibatkan pengujian rekomendasi item yang berbeda untuk mempelajari bagaimana pengguna merespons item dengan data interaksi yang sangat sedikit. Anda dapat mengonfigurasi eksplorasi dengan yang berikut ini:

- Penekanan pada penjelajahan item yang kurang relevan (bobot eksplorasi) - Konfigurasikan berapa banyak yang harus dijelajahi. Tentukan nilai desimal antara 0 hingga 1. Defaultnya adalah 0,3. Semakin dekat nilainya ke 1, semakin banyak eksplorasi. Dengan lebih banyak eksplorasi, rekomendasi mencakup lebih banyak item dengan lebih sedikit data interaksi item atau relevansi berdasarkan perilaku sebelumnya. Pada nol, tidak ada eksplorasi terjadi dan rekomendasi didasarkan pada data saat ini (relevansi).
- Batas usia item eksplorasi - Tentukan usia item maksimum dalam beberapa hari sejak interaksi terbaru di semua item dalam kumpulan data interaksi item. Ini mendefinisikan ruang lingkup

eksplorasi item berdasarkan usia item. Amazon Personalize menentukan usia item berdasarkan stempel waktu pembuatannya atau, jika data stempel waktu pembuatan tidak ada, data interaksi item. Untuk informasi selengkapnya cara Amazon Personalize menentukan usia item, lihat. Data stempel waktu pembuatan

Untuk meningkatkan item yang dipertimbangkan Amazon Personalize selama eksplorasi, masukkan nilai yang lebih besar. Minimal adalah 1 hari dan defaultnya adalah 30 hari. Rekomendasi mungkin mencakup item yang lebih tua dari potongan usia item yang Anda tentukan. Ini karena item ini relevan dengan pengguna dan eksplorasi tidak mengidentifikasinya.

Kode berikut menunjukkan cara mengonfigurasi eksplorasi saat Anda membuat pemberi rekomendasi. Contoh menggunakan nilai default.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

create_recommender_response = personalize.create_recommender(
  name = 'recommender name',
  recipeArn = 'arn:aws:personalize:::recipe/aws-vod-top-picks',
  datasetGroupArn = 'dataset group ARN',
  recommenderConfig = {"itemExplorationConfig": {"explorationWeight": "0.3",
 "explorationItemAgeCutOff": "30"}}
)

recommender_arn = create_recommender_response['recommenderArn']

print('Recommender ARN:' + recommender_arn)
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { CreateRecommenderCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});
```

```
// set the recommender's parameters
export const createRecommenderParam = {
  name: "RECOMMENDER_NAME",                    /* required */
  recipeArn: "RECIPE_ARN",                     /* required */
  datasetGroupArn: "DATASET_GROUP_ARN",        /* required */
  recommenderConfig: {
    itemExplorationConfig: {
      explorationWeight: "0.3",
      explorationItemAgeCutOff: "30"
    }
  }
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(new
 CreateRecommenderCommand(createRecommenderParam));
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Mengevaluasi rekomendasi domain Personalisasi Amazon

Anda dapat mengevaluasi kinerja pemberi rekomendasi Anda melalui metrik offline dan online. Metrik online adalah hasil empiris yang Anda amati dalam interaksi pengguna Anda dengan rekomendasi waktu nyata. Misalnya, Anda dapat merekam rasio klik-tayang pengguna saat mereka menelusuri katalog Anda. Anda bertanggung jawab untuk membuat dan merekam metrik online apa pun.

Metrik offline adalah metrik yang dihasilkan Amazon Personalize saat Anda membuat pemberi rekomendasi. Dengan metrik offline, Anda dapat mengevaluasi kinerja model yang mendukung pemberi rekomendasi Anda. Anda dapat melihat efek dari memodifikasi konfigurasi pemberi rekomendasi, dan Anda dapat membandingkan hasil dari pemberi rekomendasi yang dilatih dengan kasus penggunaan yang berbeda dengan data yang sama dalam grup kumpulan data yang sama.

Hindari membandingkan metrik rekomendasi berbeda yang dilatih dengan data yang berbeda. Perbedaan metrik mungkin berasal dari perbedaan data daripada kinerja model. Misalnya,

Anda mungkin memiliki grup kumpulan data dengan data `purchase` peristiwa jarang untuk setiap pengguna, dan grup lainnya dengan data `view` peristiwa yang kuat. Berdasarkan metrik seperti`precision at K`, pemberi rekomendasi yang dilatih tentang data peristiwa tampilan mungkin salah tampak berkinerja lebih baik karena jumlah interaksi yang lebih tinggi.

Untuk mendapatkan metrik kinerja, Amazon Personalize membagi data interaksi input menjadi satu set pelatihan dan set pengujian. Set pelatihan terdiri dari 90% pengguna Anda dan data interaksi mereka. Set pengujian terdiri dari 10% sisa pengguna dan data interaksi mereka.

Amazon Personalize kemudian membuat pemberi rekomendasi menggunakan set pelatihan. Setelah pelatihan selesai, Amazon Personalize memberi pemberi rekomendasi baru 90% tertua dari setiap data pengguna dari set pengujian sebagai input. Amazon Personalize kemudian menghitung metrik dengan membandingkan rekomendasi yang dihasilkan oleh pemberi rekomendasi dengan interaksi aktual dalam 10% terbaru dari setiap data pengguna dari set pengujian.

Topik

- Mengambil metrik
- Definisi metrik
- Contoh
- Sumber daya tambahan

Mengambil metrik

Setelah pemberi rekomendasi aktif, Anda dapat melihat metrik untuk pemberi rekomendasi di konsol Amazon Personalize atau mengambil metrik dengan memanggil operasi. DescribeRecommender

Topik

- Melihat metrik (konsol)
- Mengambil metrik ()AWS CLI
- Mengambil metrik ()AWS SDKs

Melihat metrik (konsol)

Untuk melihat metrik pemberi rekomendasi di konsol, Anda menavigasi ke halaman detail untuk pemberi rekomendasi Anda.

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Domain Anda.
3. Dari panel navigasi, pilih Rekomendasi.
4. Dari daftar pemberi rekomendasi, pilih salah satu untuk melihat metriknya.

Mengambil metrik ()AWS CLI

Kode berikut menunjukkan cara mendapatkan metrik untuk pemberi rekomendasi dengan. AWS CLI

```
aws personalize describe-recommender \
--recommender-arn recommender arn
```

Berikut ini adalah contoh keluaran metrik dari pemberi rekomendasi yang dibuat untuk pilihan Teratas untuk kasus penggunaan Anda untuk domain VIDEO_ON_DEMAND.

```
{
    "recommender": {
        "recommenderArn": "arn:aws:personalize:region:acct-id:recommender/
recommenderName",
        "datasetGroupArn": "arn:aws:personalize:region:acct-id:dataset-group/
dsGroupName",
        "name": "name123",
        "recipeArn": "arn:aws:personalize:::recipe/aws-vod-top-picks",
        "modelMetrics": {
            "coverage": 0.27,
            "mean_reciprocal_rank_at_25": 0.0379,
            "normalized_discounted_cumulative_gain_at_5": 0.0405,
            "normalized_discounted_cumulative_gain_at_10": 0.0513,
            "normalized_discounted_cumulative_gain_at_25": 0.0828,
            "precision_at_5": 0.0136,
            "precision_at_10": 0.0102,
            "precision_at_25": 0.0091,
        }
        "recommenderConfig": {},
        "creationDateTime": "2022-05-06T10:11:24.589000-07:00",
        "lastUpdatedDateTime": "2022-05-06T10:34:33.270000-07:00",
        "status": "ACTIVE",
    }
}
```

Mengambil metrik ()AWS SDKs

Kode berikut menunjukkan cara mendapatkan metrik untuk pemberi rekomendasi dengan SDK for Python (Boto3).

```
import boto3

personalize = boto3.client('personalize')

response = personalize.describe_recommender(
    recommenderArn = 'recommender_arn'
)
print(response['recommender']['modelMetrics'])
```

Berikut ini adalah contoh keluaran metrik dari pemberi rekomendasi yang dibuat untuk pilihan Teratas untuk kasus penggunaan Anda untuk domain VIDEO_ON_DEMAND.

```
{
    "recommender": {
        "recommenderArn": "arn:aws:personalize:region:acct-id:recommender/
recommenderName",
        "datasetGroupArn": "arn:aws:personalize:region:acct-id:dataset-group/
dsGroupName",
        "name": "name123",
        "recipeArn": "arn:aws:personalize:::recipe/aws-vod-top-picks",
        "modelMetrics": {
            "coverage": 0.27,
            "mean_reciprocal_rank_at_25": 0.0379,
            "normalized_discounted_cumulative_gain_at_5": 0.0405,
            "normalized_discounted_cumulative_gain_at_10": 0.0513,
            "normalized_discounted_cumulative_gain_at_25": 0.0828,
            "precision_at_5": 0.0136,
            "precision_at_10": 0.0102,
            "precision_at_25": 0.0091,
        }
        "recommenderConfig": {},
        "creationDateTime": "2022-05-06T10:11:24.589000-07:00",
        "lastUpdatedDateTime": "2022-05-06T10:34:33.270000-07:00",
        "status": "ACTIVE",
    }
}
```

Definisi metrik

Metrik yang dihasilkan Amazon Personalize untuk pemberi rekomendasi dijelaskan di bawah ini menggunakan istilah berikut:

- Rekomendasi yang relevan adalah rekomendasi untuk item yang benar-benar berinteraksi dengan pengguna. Item ini berasal dari 10% terbaru dari data interaksi setiap pengguna dari set pengujian.
- Peringkat mengacu pada posisi item yang direkomendasikan dalam daftar rekomendasi. Posisi 1 (bagian atas daftar) dianggap paling relevan bagi pengguna.

Untuk setiap metrik, angka yang lebih tinggi (mendekati 1) lebih baik. Untuk menyelam lebih dalam, lihat sumber daya yang tercantum diSumber daya tambahan.

cakupan

Nilai cakupan memberi tahu Anda proporsi item unik yang mungkin direkomendasikan Amazon Personalize dari jumlah total item unik dalam kumpulan data Interaksi dan Item. Skor cakupan yang lebih tinggi berarti Amazon Personalize merekomendasikan lebih banyak item Anda, daripada beberapa item yang sama berulang kali untuk pengguna yang berbeda. Kasus penggunaan yang menampilkan eksplorasi item, seperti Pilihan teratas untuk Anda (VIDEO_ON_DEMAND) dan Direkomendasikan untuk Anda (ECOMMERCE), memiliki cakupan yang lebih tinggi daripada yang tidak.

rata-rata peringkat timbal balik pada 25

Metrik ini memberi tahu Anda tentang kemampuan model untuk menghasilkan rekomendasi yang relevan di posisi peringkat teratas. Anda dapat memilih model dengan peringkat timbal balik rata-rata tinggi di 25 jika Anda menghasilkan hasil pencarian yang relevan untuk pengguna, dan jangan berharap pengguna memilih item yang lebih rendah dalam daftar. Misalnya, pengguna sering memilih resep memasak pertama di hasil pencarian.

Amazon Personalize menghitung metrik ini menggunakan skor peringkat timbal balik rata-rata untuk permintaan rekomendasi. Setiap skor peringkat timbal balik dihitung sebagai berikut:`1 / the rank of the highest item interacted with by the user`, di mana total peringkat yang mungkin adalah 25. Item berperingkat rendah lainnya yang berinteraksi dengan pengguna diabaikan. Jika pengguna memilih item pertama, skornya adalah 1. Jika mereka tidak memilih item apa pun, skornya adalah 0.

Misalnya, Anda mungkin menampilkan tiga pengguna yang berbeda 25 rekomendasi masing-masing:

- Jika Pengguna 1 mengklik item di peringkat 4 dan item di peringkat 10, skor peringkat timbal balik mereka adalah 1/4.
- Jika Pengguna 2 mengklik item di peringkat 2, item di peringkat 4, dan item di peringkat 12, skor peringkat timbal balik mereka adalah 1/2.
- Jika Pengguna 3 mengklik satu item di peringkat 6, skor peringkat timbal balik mereka adalah 1/6.

Peringkat timbal balik rata-rata atas semua permintaan rekomendasi (dalam hal ini 3) dihitung sebagai. `(1/4 + 1/2 + 1/6) / 3 = .3056`

keuntungan kumulatif diskon yang dinormalisasi (NDCG) di K (5, 10, atau 25)

Metrik ini memberi tahu Anda tentang seberapa baik model Anda memberi peringkat rekomendasi, di mana K adalah ukuran sampel 5, 10, atau 25 rekomendasi. Metrik ini berguna jika Anda paling tertarik dengan peringkat rekomendasi di luar hanya item peringkat tertinggi (untuk ini, lihat`mean reciprocal rank at 25`). Misalnya, skor untuk `NDCG at 10` akan berguna jika Anda memiliki aplikasi yang menampilkan hingga 10 film dalam korsel sekaligus.

Amazon Personalize menghitung NDCG dengan menetapkan bobot rekomendasi berdasarkan posisi peringkat mereka untuk setiap pengguna dalam set pengujian. Setiap rekomendasi didiskon (diberi bobot yang lebih rendah) oleh faktor yang bergantung pada posisinya. Metrik akhir adalah rata-rata semua pengguna dalam set pengujian. Keuntungan kumulatif diskon yang dinormalisasi di K mengasumsikan bahwa rekomendasi yang lebih rendah pada daftar kurang relevan daripada rekomendasi yang lebih tinggi di daftar.

Amazon Personalize menggunakan faktor pembobotan`1/log(1 + position)`, di mana bagian atas daftar adalah posisi. 1

presisi di K

Metrik ini memberi tahu Anda seberapa relevan rekomendasi model Anda berdasarkan ukuran sampel rekomendasi K (5, 10, atau 25).

Amazon Personalize menghitung metrik ini berdasarkan jumlah rekomendasi yang relevan dari rekomendasi K teratas untuk setiap pengguna dalam set pengujian, dibagi dengan K, di mana K adalah 5, 10, atau 25. Metrik akhir adalah rata-rata di semua pengguna dalam set pengujian.

Misalnya, jika Anda merekomendasikan 10 item kepada pengguna, dan pengguna berinteraksi dengan 3 item, presisi di K adalah 3 item yang diprediksi dengan benar dibagi dengan total 10 item yang direkomendasikan:`3 / 10 = .30`.

Metrik ini menghargai rekomendasi yang tepat dari item yang relevan. Semakin dekat skornya ke satu, semakin tepat modelnya.

Contoh

Berikut ini adalah contoh sederhana untuk pemberi rekomendasi yang menghasilkan daftar rekomendasi untuk pengguna tertentu. Rekomendasi kedua dan kelima cocok dengan catatan dalam data pengujian untuk pengguna ini. Ini adalah rekomendasi yang relevan. Jika `K` diatur pada`5`, metrik berikut dihasilkan untuk pengguna.

reciprocal_rank

Perhitungan: 1/2

Hasil: 0.5000

normalized_discounted_cumulative_gain_at_5

Perhitungan: (1/log (1 + 2) + 1/log (1 + 5))/(1/log (1 + 1) + 1/log (1 + 2))

Hasil: 0.6241

presisi_at_5

Perhitungan: 2/5

Hasil: 0.4000

Sumber daya tambahan

Untuk menyelami lebih dalam berbagai jenis metrik untuk sistem pemberi rekomendasi, lihat sumber daya eksternal berikut:

- MRR vs MAP vs NDCG: Metrik Evaluasi Sadar Peringkat Dan Kapan Menggunakannya
- Keuntungan Kumulatif Diskon: metrik peringkat yang harus Anda ketahui
- Recall dan Presisi di k untuk Sistem Recommender
- Metrik Evaluasi Peringkat untuk Sistem Rekomendasi

Memperbarui pemberi rekomendasi

Setelah Anda membuat pemberi rekomendasi, Anda dapat memperbarui konfigurasi pemberi rekomendasi:

- Anda dapat memperbarui kolom yang digunakan oleh pemberi rekomendasi dalam pelatihan. Jika Anda memodifikasi kolom yang digunakan saat pelatihan, Amazon Personalize secara otomatis memulai pelatihan ulang penuh model yang mendukung pemberi rekomendasi Anda. Sementara pembaruan selesai, Anda masih bisa mendapatkan rekomendasi dari pemberi rekomendasi. Rekomendasi menggunakan konfigurasi sebelumnya hingga pembaruan selesai. Untuk melacak status pembaruan ini, gunakan yang `latestRecommenderUpdate` dikembalikan dalam DescribeRecommender operasi. Jika Anda memberikan kolom yang sama dengan yang Anda berikan saat membuat pemberi rekomendasi, tidak ada pembaruan yang terjadi.
- Anda dapat memperbarui permintaan rekomendasi minimum pemberi rekomendasi per detik. Ini menentukan throughput permintaan rekomendasi dasar yang disediakan oleh Amazon Personalize. Nilai tinggi akan meningkatkan tagihan Anda. Sebaiknya mulai dengan 1. Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan seperlunya. Untuk informasi selengkapnya, lihat Permintaan rekomendasi minimum per detik dan auto-scaling.
- Untuk Pilihan teratas untuk Anda dan Direkomendasikan untuk kasus penggunaan Anda, Anda dapat memperbarui konfigurasi eksplorasi dengan menyesuaikan penekanan pada penjelajahan item yang relevan dan batas usia item eksplorasi. Untuk informasi tentang eksplorasi, lihat bagian untuk kasus penggunaan Anda diMemilih kasus penggunaan.

Anda dapat memperbarui pemberi rekomendasi dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau. AWS SDKs

Topik

- Memperbarui pemberi rekomendasi (Amazon Personalize console)
- Memperbarui pemberi rekomendasi ()AWS CLI
- Memperbarui pemberi rekomendasi ()AWS SDKs

Memperbarui pemberi rekomendasi (Amazon Personalize console)

Setelah Anda membuat pemberi rekomendasi, Anda dapat memperbaruinya. Anda dapat memperbarui kolom yang digunakan pemberi rekomendasi dalam pelatihan dan permintaan

rekomendasi minimum pemberi rekomendasi per detik. Untuk Pilihan teratas untuk Anda dan Direkomendasikan untuk kasus penggunaan Anda, Anda dapat memperbarui konfigurasi eksplorasi. Untuk memperbarui pemberi rekomendasi dengan konsol, lakukan hal berikut.

Untuk memperbarui konfigurasi pemberi rekomendasi (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Domain Anda.
3. Dari panel navigasi, pilih Rekomendasi.
4. Pada halaman Rekomendasi, pilih pemberi rekomendasi yang ingin Anda perbarui.
5. Dalam konfigurasi Recommender pilih Edit.
6. Ubah konfigurasi pemberi rekomendasi dan pilih Perbarui. Untuk informasi tentang opsi konfigurasi yang berbeda, lihatMembuat rekomendasi (konsol).

Memperbarui pemberi rekomendasi ()AWS CLI

Untuk memperbarui pemberi rekomendasi dengan AWS CLI, gunakan `update-recommender` perintah. Berikan Nama Sumber Daya Amazon (ARN) untuk pemberi rekomendasi dan konfigurasi yang diperbarui. Kode berikut menunjukkan cara memperbarui kolom yang digunakan oleh pemberi rekomendasi untuk pelatihan.

```
aws personalize update-recommender \
--dataset-group-arn dataset group ARN \
--recommender-config "{\"trainingDataConfig\": {\"excludedDatasetColumns\":
 { \"datasetType\" : [ \"column1Name\", \"column2Name\"]}}}"
```

Jika Anda memodifikasi kolom yang digunakan dalam pelatihan, Amazon Personalize secara otomatis memulai pelatihan ulang penuh model yang mendukung pemberi rekomendasi Anda. Sementara pembaruan selesai, Anda masih bisa mendapatkan rekomendasi dari pemberi rekomendasi. Rekomendasi menggunakan konfigurasi sebelumnya hingga pembaruan selesai. Untuk melacak status pembaruan ini, gunakan yang `latestRecommenderUpdate` dikembalikan dalam DescribeRecommender operasi.

Untuk informasi selengkapnya tentang berbagai konfigurasi yang dapat Anda ubah, lihatRecommenderConfig.

Memperbarui pemberi rekomendasi ()AWS SDKs

Untuk memperbarui pemberi rekomendasi dengan AWS, gunakan UpdateRecommender operasi. Berikan Nama Sumber Daya Amazon (ARN) untuk pemberi rekomendasi dan tentukan konfigurasi baru. Kode berikut menunjukkan cara memperbarui kolom yang digunakan oleh pemberi rekomendasi untuk pelatihan.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

update_recommender_response = personalize.update_recommender(
  recommenderArn = 'dataset group ARN',
  recommenderConfig = {
    "trainingDataConfig": {
      "excludedDatasetColumns": {
        "datasetType": ["COLUMN_A", "COLUMN_B"]
      }
    }
  }
)
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { UpdateRecommenderCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

// set the request's parameters
export const updateRecommenderParam = {
  recommenderArn: "RECOMMENDER_ARN", /* required */
  recommenderConfig: {
    trainingDataConfig: {
      excludedDatasetColumns: {
        "DATASET_TYPE": ["COLUMN_A", "COLUMN_B"]
```

```
      }
    }
  }
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(new
 UpdateRecommenderCommand(updateRecommenderParam));
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Jika Anda memodifikasi kolom yang digunakan dalam pelatihan di `excludedDatasetColumns` bagian`recommenderConfig`, Amazon Personalize secara otomatis memulai pelatihan ulang penuh model yang mendukung pemberi rekomendasi Anda. Sementara pembaruan selesai, Anda masih bisa mendapatkan rekomendasi dari pemberi rekomendasi. Rekomendasi menggunakan konfigurasi sebelumnya hingga pembaruan selesai. Untuk melacak status pembaruan ini, gunakan yang `latestRecommenderUpdate` dikembalikan dalam DescribeRecommender operasi.

Untuk informasi selengkapnya tentang berbagai konfigurasi yang dapat Anda ubah, lihatRecommenderConfig.

Menghentikan pemberi rekomendasi

Setelah pemberi rekomendasi Anda aktif, Anda dapat menghentikan pemberi rekomendasi dan memulainya nanti. Dengan cara ini, Anda dapat menjeda penagihan pemberi rekomendasi dan hanya membayarnya saat Anda menggunakannya. Misalnya, Anda mungkin perlu mendapatkan rekomendasi hanya pada hari-hari tertentu dalam seminggu. Anda dapat menghentikan pemberi rekomendasi pada hari-hari Anda tidak membutuhkannya, dan kemudian memulai pemberi rekomendasi pada hari-hari yang Anda lakukan.

Setelah Anda menghentikan pemberi rekomendasi, Anda tidak dapat menggunakannya untuk mendapatkan rekomendasi. Menghentikan pemberi rekomendasi menghentikan penagihan dan pelatihan ulang pemberi rekomendasi. Namun, menghentikan pemberi rekomendasi tidak

menghapus pemberi rekomendasi. Anda dapat memulai ulang kapan saja dan melanjutkan mendapatkan rekomendasi. Memulai pemberi rekomendasi tidak membuat pemberi rekomendasi baru dengan data Anda. Sebaliknya, ia melanjutkan penagihan dan pelatihan ulang pemberi rekomendasi setiap 7 hari.

Anda dapat menghentikan dan memulai pemberi rekomendasi dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(),. AWS SDKs

Merekomendasikan menyatakan

Saat Anda menghentikan pemberi rekomendasi, status pemberi rekomendasi berubah dari ACTIVE ke INACTIVE dalam urutan berikut:

AKTIF> BERHENTI TERTUNDA > BERHENTI DALAM PROSES > TIDAK AKTIF

Saat Anda memulai pemberi rekomendasi, status pemberi rekomendasi berubah dari INACTIVE ke ACTIVE dalam urutan berikut:

TIDAK AKTIF> MULAI TERTUNDA > MULAI DALAM PROSES > AKTIF

Topik

- Menghentikan pemberi rekomendasi (konsol)
- Menghentikan pemberi rekomendasi ()AWS CLI
- Menghentikan pemberi rekomendasi ()AWS SDKs

Menghentikan pemberi rekomendasi (konsol)

Anda dapat menghentikan pemberi rekomendasi dari halaman detail untuk pemberi rekomendasi di konsol Amazon Personalize.

Untuk menghentikan pemberi rekomendasi

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Domain Anda.
3. Dari panel navigasi, pilih Rekomendasi.
4. Pada halaman Rekomendasi, pilih pemberi rekomendasi yang ingin Anda hentikan.

5. Pada halaman detail untuk pemberi rekomendasi, pilih Stop recommended di kanan atas dan konfirmasi di jendela yang ditampilkan. Ketika status pemberi rekomendasi tidak aktif, pemberi rekomendasi Anda telah berhenti. Anda dapat memulainya lagi dari halaman yang sama.

Menghentikan pemberi rekomendasi ()AWS CLI

Untuk menghentikan pemberi rekomendasi aktif dengan AWS CLI, gunakan `stop-recommender` perintah, yang menggunakan operasi StopRecommender API, dan berikan Nama Sumber Daya Amazon (ARN) untuk pemberi rekomendasi. Untuk memulai ulang, Anda dapat menggunakan `start-recommender` perintah, yang menggunakanStartRecommender. Kode berikut menunjukkan cara menghentikan pemberi rekomendasi:

```
aws personalize stop-recommender --recommender-arn "recommender arn"
```

Menghentikan pemberi rekomendasi ()AWS SDKs

Untuk menghentikan pemberi rekomendasi aktif dengan AWS SDKs, gunakan operasi StopRecommender API, dan berikan Nama Sumber Daya Amazon (ARN) untuk pemberi rekomendasi. Untuk memulai ulang, Anda menggunakanStartRecommender. Kode berikut menunjukkan cara menghentikan pemberi rekomendasi:

SDK for Python (Boto3)

Untuk menghentikan pemberi rekomendasi aktif dengan SDK for Python (Boto3), `stop_recommender` gunakan metode ini dan berikan Nama Sumber Daya Amazon (ARN) untuk pemberi rekomendasi sebagai berikut.

```
import boto3
personalize = boto3.client('personalize')

stop_recommender_response = personalize.stop_recommender(
    recommenderArn = "recommenderARN"
)
print(stop_recommender_response)
```

SDK for Java 2.x

Untuk menghentikan pemberi rekomendasi aktif dengan SDK for Java 2.x, gunakan `stopRecommender` metode ini dan berikan ARN untuk pemberi rekomendasi sebagai berikut.

```
public static void stopRecommender(PersonalizeClient personalizeClient,
                                                String datasetGroupArn) {

    try {

        StopRecommenderRequest stopRecommenderRequest =
 StopRecommenderRequest.builder()
                .recommenderArn(recommenderArn)
                .build();
        personalizeClient.stopRecommender(stopRecommenderRequest);
    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { StopRecommenderCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

// set the request params
export const stopRecommenderParam = {
  recommenderArn: "RECOMMENDER_ARN" /* required */
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new StopRecommenderCommand(stopRecommenderParam)
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
```

```
run();
```

Sumber daya khusus untuk melatih dan menerapkan model Amazon Personalize

Jika Anda menggunakan resep khusus, setelah mengimpor data, Anda siap membuat sumber daya khusus untuk pelatihan dan menerapkan model Amazon Personalize. Anda menggunakan sumber daya ini untuk mendapatkan rekomendasi. Saat membuat sumber daya khusus, Anda melakukan hal berikut:

1. Buat dan konfigurasikan solusi: Sesuaikan parameter solusi dan hiperparameter khusus resep sehingga model memenuhi kebutuhan bisnis spesifik Anda. Secara default, versi solusi baru menggunakan pelatihan otomatis untuk membuat versi solusi pada frekuensi yang dapat dikonfigurasi. Frekuensi default adalah setiap 7 hari. Untuk informasi selengkapnya tentang mengonfigurasi solusi, lihatMengkonfigurasi solusi. Untuk informasi selengkapnya tentang resep khusus di Amazon Personalize, lihat. Memilih resep
2. Buat versi solusi (untuk solusi yang tidak menggunakan pelatihan otomatis): Untuk solusi yang menggunakan pelatihan otomatis, pembuatan versi solusi dimulai secara otomatis setelah solusi Anda aktif. Untuk solusi yang menggunakan pelatihan manual, Anda membuat versi solusi secara manual. Versi solusi menghasilkan rekomendasi Amazon Personalize atau segmen pengguna. Untuk informasi selengkapnya tentang membuat versi solusi secara manual, lihatMembuat versi solusi secara manual. Untuk menghentikan pembuatan versi solusi, navigasikan ke halaman detail versi solusi dan pilih Berhenti. Untuk informasi selengkapnya, lihat Menghentikan pembuatan versi solusi.
3. Evaluasi versi solusi — Gunakan metrik yang dihasilkan Amazon Personalize dari versi solusi baru untuk mengevaluasi kinerja model. Lihat Mengevaluasi versi solusi Amazon Personalize dengan metrik.
4. Menerapkan versi solusi dengan kampanye (hanya untuk rekomendasi waktu nyata): Buat kampanye untuk menerapkan versi solusi Anda. Anda menggunakan kampanye saat meminta rekomendasi waktu nyata. Jika Anda mendapatkan rekomendasi batch, Anda tidak perlu membuat kampanye. Untuk informasi selengkapnya, lihat Menerapkan versi solusi Amazon Personalize dengan kampanye. Jika ingin mengubah setelan kampanye yang ada, seperti mengaktifkan metadata dalam rekomendasi, Anda harus memperbarui kampanye. Lihat informasi yang lebih lengkap di Memperbarui konfigurasi kampanye Amazon Personalize.

Topik

- Mengonfigurasi solusi khusus di Amazon Personalize
- Memperbarui solusi untuk mengubah konfigurasi pelatihan otomatisnya
- Membuat versi solusi secara manual
- Menghentikan pembuatan versi solusi
- Mengevaluasi versi solusi Amazon Personalize dengan metrik
- Menerapkan versi solusi Amazon Personalize dengan kampanye
- Memperbarui konfigurasi kampanye Amazon Personalize

Mengonfigurasi solusi khusus di Amazon Personalize

Setelah Anda selesai mengimpor data, Anda siap untuk membuat solusi. Solusi mengacu pada kombinasi resep Amazon Personalize, parameter pelatihan yang disesuaikan, dan satu atau beberapa versi solusi. Versi solusi mengacu pada model pembelajaran mesin yang terlatih.

Secara default, semua solusi baru menggunakan pelatihan otomatis untuk membuat versi solusi baru setiap 7 hari. Pelatihan otomatis hanya terjadi jika Anda mengimpor data interaksi massal atau waktu nyata sejak pelatihan terakhir. Ini termasuk interaksi item atau, untuk solusi yang menggunakan Next-Best-Action resep, data interaksi tindakan. Pelatihan otomatis berlanjut hingga Anda menghapus solusinya. Untuk informasi selengkapnya, lihat Mengkonfigurasi pelatihan otomatis.

Jika Anda memiliki solusi yang ada, Anda dapat menggunakan konsol Amazon Personalize untuk mengkloning solusi. Saat Anda mengkloning solusi, Anda dapat menggunakan konfigurasi solusi yang ada sebagai titik awal, seperti resep dan hiperparameter, dan membuat perubahan apa pun. Untuk informasi selengkapnya, lihat Mengkloning solusi (konsol).

Anda dapat membuat dan mengonfigurasi solusi dengan menggunakan konsol, AWS Command Line Interface (AWS CLI), atau AWS SDKs. Setelah membuat solusi, Anda dapat melihat detail konfigurasinya di halaman detail solusi di konsol Amazon Personalize, atau dengan operasi. DescribeSolution

Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Topik

- Menciptakan solusi
- Mengkonfigurasi pelatihan otomatis
- Mengkonfigurasi kolom yang digunakan saat pelatihan
- Mengoptimalkan solusi untuk tujuan tambahan
- Mengoptimalkan solusi dengan konfigurasi peristiwa
- Hyperparameters dan HPO
- Memilih data interaksi item yang digunakan untuk pelatihan
- Mengkloning solusi (konsol)

Menciptakan solusi

Anda dapat membuat solusi kustom dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs Berikut ini mencakup langkah-langkah terperinci untuk membuat solusi dengan konsol Amazon Personalize dan contoh kode yang menunjukkan cara membuat solusi hanya dengan bidang yang diperlukan.

Topik

- Membuat solusi (konsol)
- Membuat solusi (AWS CLI)
- Membuat solusi (AWS SDKs)

Membuat solusi (konsol)

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Untuk membuat solusi di konsol, pilih grup kumpulan data Anda, lalu tentukan nama solusi, resep, dan konfigurasi pelatihan opsional.

Untuk mengkonfigurasi solusi (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah, dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
3. Pada halaman Ikhtisar, untuk Langkah 3, lakukan salah satu hal berikut:
   - Jika Anda membuat grup kumpulan data Domain, pilih Gunakan sumber daya khusus, dan pilih Buat solusi.
   - Jika Anda membuat grup kumpulan data kustom, pilih Buat solusi.
4. Untuk nama Solusi, tentukan nama untuk solusi Anda.
5. Untuk tipe Solusi, pilih jenis solusi yang ingin Anda buat. Jenis yang Anda pilih menentukan resep apa yang tersedia.
   - Pilih Rekomendasi item untuk mendapatkan rekomendasi item untuk pengguna Anda. Misalnya, rekomendasi film yang dipersonalisasi.
   - Pilih Rekomendasi tindakan untuk mendapatkan rekomendasi tindakan bagi pengguna Anda. Misalnya, buat tindakan terbaik berikutnya untuk pengguna, seperti mengunduh aplikasi Anda.
   - Pilih Segmentasi pengguna untuk mendapatkan segmen pengguna (grup pengguna) berdasarkan data item Anda.
6. Untuk Resep, pilih resep (lihatMemilih resep).
7. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya
8. Pilih Berikutnya.
9. Pada halaman konfigurasi Pelatihan, sesuaikan solusi untuk memenuhi kebutuhan bisnis Anda.
   - Dalam Pelatihan otomatis, pilih apakah solusinya menggunakan pelatihan otomatis. Jika Anda menggunakan pelatihan otomatis, Anda dapat mengubah`Automatic training frequency`. Frekuensi pelatihan default adalah setiap 7 hari.

     Kami merekomendasikan menggunakan pelatihan otomatis. Itu membuatnya lebih mudah bagi Anda untuk mempertahankan relevansi rekomendasi. Frekuensi pelatihan Anda tergantung pada kebutuhan bisnis Anda, resep yang Anda gunakan, dan seberapa sering Anda mengimpor data. Untuk informasi selengkapnya, lihat Mengkonfigurasi pelatihan otomatis. Untuk informasi tentang menjaga relevansi, lihatMempertahankan relevansi rekomendasi.

- Dalam konfigurasi Hyperparameter, konfigurasikan opsi hyperparameter apa pun berdasarkan resep dan kebutuhan bisnis Anda. Resep yang berbeda menggunakan hyperparameters yang berbeda. Untuk hiperparameter yang tersedia untuk Anda, lihat masing-masing resep diMemilih resep.
- Di Kolom untuk pelatihan, jika resep Anda menghasilkan rekomendasi item atau segmen pengguna, pilih kolom yang dipertimbangkan Amazon Personalize saat membuat versi solusi. Untuk informasi selengkapnya, lihat Mengkonfigurasi kolom yang digunakan saat pelatihan.
- Dalam konfigurasi Peristiwa, jika kumpulan data interaksi Item Anda memiliki EVENT_TYPE atau kolom EVENT_TYPE dan EVENT_VALUE, gunakan kolom Tipe peristiwa dan ambang nilai Peristiwa secara opsional untuk memilih data interaksi item yang digunakan Amazon Personalize saat melatih model. Untuk informasi selengkapnya, lihat Memilih data interaksi item yang digunakan untuk pelatihan.

  Jika Anda memiliki beberapa jenis acara dan menggunakan resep User-Personalization-v 2 atau Personalized-Ranking-v 2 resep, Anda juga dapat menentukan bobot yang berbeda untuk jenis yang berbeda. Misalnya, Anda dapat mengonfigurasi solusi untuk memberikan bobot lebih untuk membeli acara daripada peristiwa klik. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi dengan konfigurasi peristiwa.
- Jika Anda menggunakan salah satu Resep Peringkat yang Dipersonalisasi resep Resep Personalisasi Pengguna atau, secara opsional tentukan Objective dan pilih sensitivitas Objective untuk mengoptimalkan solusi Anda untuk tujuan selain relevansi. Sensitivitas obyektif mengonfigurasi cara Amazon Personalize menyeimbangkan item yang direkomendasikan berdasarkan tujuan Anda dibandingkan dengan relevansi melalui data interaksi. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi untuk tujuan tambahan.

10. Pilih Berikutnya dan tinjau detail solusinya. Anda tidak dapat mengubah konfigurasi solusi Anda setelah Anda membuatnya.
11. Pilih Buat solusi. Setelah Anda membuat solusi, Amazon Personalize mulai membuat versi solusi pertama Anda dalam waktu satu jam. Saat pelatihan dimulai, Anda dapat memantaunya di bagian Versi solusi di halaman detail untuk solusi Anda. Versi solusi yang dibuat secara otomatis memiliki jenis Pelatihan OTOMATIS.

    Ketika versi solusi AKTIF, Anda siap menggunakannya untuk mendapatkan rekomendasi. Bagaimana Anda menggunakan versi solusi aktif tergantung pada bagaimana Anda mendapatkan rekomendasi:

- Untuk rekomendasi real-time, Anda menerapkan versi solusi AKTIF dengan kampanye Amazon Personalize. Anda menggunakan kampanye untuk mendapatkan rekomendasi bagi pengguna Anda. Lihat Menerapkan versi solusi Amazon Personalize dengan kampanye.
- Untuk rekomendasi batch, Anda menentukan versi solusi ACTIVE saat membuat pekerjaan inferensi batch atau pekerjaan segmen batch. Lihat Mendapatkan rekomendasi item batch atau Mendapatkan segmen pengguna batch.

Membuat solusi (AWS CLI)

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Untuk membuat solusi dengan AWS CLI, gunakan `create-solution` perintah. Perintah ini menggunakan operasi CreateSolution API. Kode berikut menunjukkan cara membuat solusi yang menggunakan pelatihan otomatis. Secara otomatis membuat versi solusi baru setiap lima hari.

Untuk menggunakan kode, perbarui kode untuk memberi solusi nama, tentukan Nama Sumber Daya Amazon (ARN) grup kumpulan data Anda, ubah frekuensi pelatihan secara opsional, dan tentukan ARN resep yang akan digunakan. Untuk informasi tentang resep, lihatMemilih resep.

```
aws personalize create-solution \
--name solution name \
--dataset-group-arn dataset group ARN \
--recipe-arn recipe ARN \
--perform-auto-training \
--solution-config "{\"autoTrainingConfig\": {\"schedulingExpression\": \"rate(5
 days)\"}}"
```

- Kami menyarankan Anda menggunakan pelatihan otomatis. Ini memudahkan Anda untuk mempertahankan dan meningkatkan relevansi rekomendasi. Secara default, semua solusi baru menggunakan pelatihan otomatis. Frekuensi pelatihan default adalah setiap 7 hari. Frekuensi

pelatihan Anda tergantung pada kebutuhan bisnis Anda, resep yang Anda gunakan, dan seberapa sering Anda mengimpor data. Untuk informasi selengkapnya, lihat Mengkonfigurasi pelatihan otomatis.

- Bergantung pada resep Anda, Anda dapat memodifikasi kode untuk mengonfigurasi properti khusus resep dan hiperparameter (lihatHyperparameters dan HPO), mengonfigurasi kolom yang digunakan untuk pelatihan (lihatMengkonfigurasi kolom yang digunakan saat pelatihan ()AWS CLI), atau memfilter data interaksi item yang digunakan untuk pelatihan (lihatMemilih data interaksi item yang digunakan untuk pelatihan).
- Jika Anda menggunakan Resep Peringkat yang Dipersonalisasi resep Resep Personalisasi Pengguna atau, Anda dapat mengoptimalkan solusi Anda untuk suatu tujuan, selain relevansi. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi untuk tujuan tambahan.

Setelah Anda membuat solusi, catat solusi ARN untuk penggunaan masa depan. Dengan pelatihan otomatis, pembuatan versi solusi dimulai dalam waktu satu jam setelah solusi AKTIF. Jika Anda secara manual membuat versi solusi dalam satu jam, solusi melewatkan pelatihan otomatis pertama. Setelah pelatihan dimulai, Anda bisa mendapatkan Amazon Resource Name (ARN) versi solusi dengan operasi ListSolutionVersionsAPI. Untuk mendapatkan statusnya, gunakan operasi DescribeSolutionVersionAPI.

Ketika versi solusi AKTIF, Anda siap menggunakannya untuk mendapatkan rekomendasi. Bagaimana Anda menggunakan versi solusi aktif tergantung pada bagaimana Anda mendapatkan rekomendasi:

- Untuk rekomendasi real-time, Anda menerapkan versi solusi AKTIF dengan kampanye Amazon Personalize. Anda menggunakan kampanye untuk mendapatkan rekomendasi bagi pengguna Anda. Lihat Menerapkan versi solusi Amazon Personalize dengan kampanye.
- Untuk rekomendasi batch, Anda menentukan versi solusi ACTIVE saat membuat pekerjaan inferensi batch atau pekerjaan segmen batch. Lihat Mendapatkan rekomendasi item batch atau Mendapatkan segmen pengguna batch.

Membuat solusi (AWS SDKs)

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk

mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Untuk membuat solusi dengan AWS SDKs, gunakan operasi CreateSolution API. Kode berikut menunjukkan cara membuat solusi yang menggunakan pelatihan otomatis. Secara otomatis membuat versi solusi baru setiap lima hari.

Untuk menggunakan kode, perbarui kode untuk memberi solusi nama, tentukan Nama Sumber Daya Amazon (ARN) grup kumpulan data Anda, ubah frekuensi pelatihan secara opsional, dan tentukan ARN resep yang ingin Anda gunakan. Untuk informasi tentang resep, lihatMemilih resep.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

create_solution_response = personalize.create_solution(
  name = 'solution name',
  recipeArn = 'recipe ARN',
  datasetGroupArn = 'dataset group ARN',
  performAutoTraining = True,
  solutionConfig = {
    "autoTrainingConfig": {
      "schedulingExpression": "rate(5 days)"
    }
  }
)
solution_arn = create_solution_response['solutionArn']
print('solution_arn: ', solution_arn)
```

SDK for JavaScript v3

```

import {
  CreateSolutionCommand,
  PersonalizeClient,
} from "@aws-sdk/client-personalize";

// create client
```

```
const personalizeClient = new PersonalizeClient({ region: "REGION" });

// set the solution parameters
export const solutionParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  recipeArn: "RECIPE_ARN" /* required */,
  name: "SOLUTION_NAME" /* required */,
  performAutoTraining: true /* optional, default is true */,
  solutionConfig: {
    autoTrainingConfig: {
      schedulingExpression:
        "rate(5 days)" /* optional, default is every 7 days */,
    },
  },
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSolutionCommand(solutionParam)
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Kami menyarankan Anda menggunakan pelatihan otomatis. Ini memudahkan Anda untuk mempertahankan dan meningkatkan relevansi rekomendasi. Secara default, semua solusi baru menggunakan pelatihan otomatis. Frekuensi pelatihan default adalah setiap 7 hari. Frekuensi pelatihan Anda tergantung pada kebutuhan bisnis Anda, resep yang Anda gunakan, dan seberapa sering Anda mengimpor data. Untuk informasi selengkapnya, lihat Mengkonfigurasi pelatihan otomatis.
- Bergantung pada resep Anda, Anda dapat memodifikasi kode untuk mengonfigurasi properti khusus resep dan hiperparameter (lihatHyperparameters dan HPO), mengonfigurasi kolom yang digunakan untuk pelatihan (lihatMengkonfigurasi kolom yang digunakan saat pelatihan ()AWS SDKs), atau memfilter data interaksi item yang digunakan untuk pelatihan (lihatMemilih data interaksi item yang digunakan untuk pelatihan).

- Jika Anda menggunakan Resep Peringkat yang Dipersonalisasi resep Resep Personalisasi Pengguna atau, Anda dapat mengoptimalkan solusi Anda untuk suatu tujuan, selain relevansi. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi untuk tujuan tambahan.

Setelah Anda membuat solusi, catat solusi ARN untuk penggunaan masa depan. Dengan pelatihan otomatis, pembuatan versi solusi dimulai dalam waktu satu jam setelah solusi AKTIF. Jika Anda secara manual membuat versi solusi dalam satu jam, solusi melewatkan pelatihan otomatis pertama. Setelah pelatihan dimulai, Anda bisa mendapatkan Amazon Resource Name (ARN) versi solusi dengan operasi ListSolutionVersionsAPI. Untuk mendapatkan statusnya, gunakan operasi DescribeSolutionVersionAPI.

Anda dapat menggunakan kode Python berikut untuk menunggu pelatihan otomatis dimulai. `wait_for_training_to_start`Metode mengembalikan ARN dari versi solusi pertama.

```
import time
import boto3


def wait_for_training_to_start(new_solution_arn):
    max_time = time.time() + 3 * 60 * 60    # 3 hours
    while time.time() < max_time:
        list_solution_versions_response = personalize.list_solution_versions(
            solutionArn=new_solution_arn
        )
        solution_versions = list_solution_versions_response.get('solutionVersions', [])
        if solution_versions:
            new_solution_version_arn = solution_versions[0]['solutionVersionArn']
            print(f"Solution version ARN: {new_solution_version_arn}")
            return new_solution_version_arn
        else:
            print(f"Training hasn't started yet. Training will start within the next
 hour.")
            time.sleep(60)


personalize = boto3.client('personalize')

solution_arn = "solution_arn"
solution_version_arn = wait_for_training_to_start(solution_arn)
```

Ketika versi solusi AKTIF, Anda siap menggunakannya untuk mendapatkan rekomendasi. Bagaimana Anda menggunakan versi solusi aktif tergantung pada bagaimana Anda mendapatkan rekomendasi:

- Untuk rekomendasi real-time, Anda menerapkan versi solusi AKTIF dengan kampanye Amazon Personalize. Anda menggunakan kampanye untuk mendapatkan rekomendasi bagi pengguna Anda. Lihat Menerapkan versi solusi Amazon Personalize dengan kampanye.
- Untuk rekomendasi batch, Anda menentukan versi solusi ACTIVE saat membuat pekerjaan inferensi batch atau pekerjaan segmen batch. Lihat Mendapatkan rekomendasi item batch atau Mendapatkan segmen pengguna batch.

Mengkonfigurasi pelatihan otomatis

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Saat Anda membuat solusi, Anda dapat mengonfigurasi apakah solusinya menggunakan pelatihan otomatis. Anda juga dapat mengonfigurasi frekuensi pelatihan. Misalnya, Anda dapat mengonfigurasi solusi untuk membuat versi solusi baru setiap lima hari.

Secara default, semua solusi baru menggunakan pelatihan otomatis untuk membuat versi solusi baru setiap 7 hari. Pelatihan otomatis hanya terjadi jika Anda mengimpor data interaksi massal atau waktu nyata sejak pelatihan terakhir. Ini termasuk interaksi item atau, untuk solusi yang menggunakan Next-Best-Action resep, data interaksi tindakan. Pelatihan otomatis berlanjut hingga Anda menghapus solusinya.

Kami menyarankan Anda menggunakan pelatihan otomatis. Itu membuat mempertahankan solusi Anda lebih mudah. Ini menghapus pelatihan manual yang diperlukan untuk solusi untuk belajar dari data terbaru Anda. Tanpa pelatihan otomatis, Anda harus membuat versi solusi baru secara manual agar solusi dapat dipelajari dari data terbaru Anda. Ini dapat menghasilkan rekomendasi basi dan tingkat konversi yang lebih rendah. Untuk informasi selengkapnya tentang mempertahankan rekomendasi Amazon Personalize, lihat. Mempertahankan relevansi rekomendasi

Anda dapat mengonfigurasi pelatihan otomatis dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs Untuk langkah-langkah mengonfigurasi pelatihan otomatis dengan konsol, lihatMembuat solusi (konsol).

Setelah Anda membuat solusi, catat solusi ARN untuk penggunaan masa depan. Dengan pelatihan otomatis, pembuatan versi solusi dimulai dalam waktu satu jam setelah solusi AKTIF. Jika Anda membuat versi solusi secara manual dalam satu jam, solusinya melewatkan pelatihan otomatis pertama. Setelah pelatihan dimulai, Anda bisa mendapatkan Amazon Resource Name (ARN) versi solusi dengan operasi ListSolutionVersionsAPI. Untuk mendapatkan statusnya, gunakan operasi DescribeSolutionVersionAPI.

Topik

- Pedoman dan persyaratan
- Mengkonfigurasi pelatihan otomatis ()AWS CLI
- Mengkonfigurasi pelatihan otomatis () SDKs

Pedoman dan persyaratan

Berikut ini adalah pedoman dan persyaratan untuk pelatihan otomatis:

- Pelatihan otomatis hanya terjadi jika Anda mengimpor data interaksi massal atau waktu nyata sejak pelatihan terakhir. Ini termasuk interaksi item atau, untuk solusi yang menggunakan Next-Best-Action resep, data interaksi tindakan.
- Setiap pelatihan mempertimbangkan semua data dalam grup dataset Anda yang Anda sertakan dalam pelatihan. Untuk informasi tentang mengonfigurasi kolom yang digunakan dalam pelatihan, lihatMengkonfigurasi kolom yang digunakan saat pelatihan.
- Anda masih dapat membuat versi solusi secara manual.
- Pelatihan otomatis dimulai dalam waktu satu jam setelah solusi Anda aktif. Jika Anda membuat versi solusi secara manual dalam satu jam, solusinya melewatkan pelatihan otomatis pertama.
- Penjadwalan pelatihan didasarkan pada tanggal mulai pelatihan. Misalnya, jika versi solusi pertama Anda memulai pelatihan pada pukul 19:00, dan Anda menggunakan pelatihan mingguan, versi solusi berikutnya akan memulai pelatihan seminggu kemudian pada pukul 7:00 malam.
- Untuk semua resep, kami merekomendasikan setidaknya frekuensi pelatihan mingguan. Anda dapat menentukan frekuensi pelatihan antara 1 dan 30 hari. Defaultnya adalah setiap 7 hari.
  - Jika Anda menggunakan User-Personalization-v 2, Personalisasi Pengguna, atau Next-Best-Action, solusi akan diperbarui secara otomatis untuk mempertimbangkan item atau tindakan baru

untuk rekomendasi. Pembaruan otomatis tidak sama dengan pelatihan otomatis. Pembaruan otomatis tidak membuat versi solusi yang sama sekali baru, dan model tidak belajar dari data terbaru Anda. Untuk mempertahankan solusi Anda, frekuensi pelatihan Anda harus tetap setidaknya setiap minggu. Untuk informasi selengkapnya tentang pembaruan otomatis, termasuk pedoman dan persyaratan tambahan, lihatPembaruan otomatis.

- Jika Anda menggunakan Trending-Now, Amazon Personalize secara otomatis mengidentifikasi item tren teratas dalam data interaksi Anda selama interval waktu yang dapat dikonfigurasi. Trending-Now dapat merekomendasikan item yang ditambahkan sejak pelatihan terakhir melalui data interaksi massal atau streaming. Frekuensi pelatihan Anda harus tetap setidaknya setiap minggu. Untuk informasi selengkapnya, lihat Resep Trend-Now.
- Jika Anda tidak menggunakan resep dengan pembaruan otomatis atau resep Trending-Now, Amazon Personalize mempertimbangkan item baru untuk rekomendasi hanya setelah pelatihan berikutnya. Misalnya, jika Anda menggunakan resep Barang Serupa dan menambahkan item baru setiap hari, Anda harus menggunakan frekuensi pelatihan otomatis harian agar item ini muncul dalam rekomendasi pada hari yang sama.

Mengkonfigurasi pelatihan otomatis ()AWS CLI

Kode berikut menunjukkan cara membuat solusi yang secara otomatis membuat versi solusi setiap lima hari. Untuk mematikan pelatihan otomatis, atur `perform-auto-training` kefalse.

Untuk mengubah frekuensi pelatihan, Anda dapat memodifikasi `schedulingExpression` di`autoTrainingConfig`. Ekspresi harus dalam `rate(`*`value unit`*`)` format. Untuk nilainya, tentukan angka antara 1 dan 30. Untuk unit, tentukan `day` atau`days`.

Untuk penjelasan lengkap tentang `create-solution` perintah tersebut, lihatMembuat solusi (AWS CLI).

```
aws personalize create-solution \
--name solution name \
--dataset-group-arn dataset group ARN \
--recipe-arn recipe ARN \
--perform-auto-training \
--solution-config "{\"autoTrainingConfig\": {\"schedulingExpression\": \"rate(5
 days)\"}}"
```

Mengkonfigurasi pelatihan otomatis () SDKs

Kode berikut menunjukkan kepada Anda cara membuat solusi dengan pelatihan otomatis dengan AWS SDKs. Solusi secara otomatis membuat versi solusi setiap lima hari. Untuk mematikan pelatihan otomatis, atur `performAutoTraining` ke`false`.

Untuk mengubah frekuensi pelatihan, Anda dapat memodifikasi `schedulingExpression` di`autoTrainingConfig`. Ekspresi harus dalam `rate(`*`value unit`*`)` format. Untuk nilainya, tentukan angka antara 1 dan 30. Untuk unit, tentukan `day` atau`days`.

Untuk penjelasan selengkapnya tentang operasi CreateSolution API, lihatMembuat solusi (AWS SDKs).

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

create_solution_response = personalize.create_solution(
  name = 'solution name',
  recipeArn = 'recipe ARN',
  datasetGroupArn = 'dataset group ARN',
  performAutoTraining = True,
  solutionConfig = {
    "autoTrainingConfig": {
      "schedulingExpression": "rate(5 days)"
    }
  }
)
solution_arn = create_solution_response['solutionArn']
print('solution_arn: ', solution_arn)
```

SDK for JavaScript v3

```

import {
  CreateSolutionCommand,
  PersonalizeClient,
} from "@aws-sdk/client-personalize";

// create client
```

```
const personalizeClient = new PersonalizeClient({ region: "REGION" });

// set the solution parameters
export const solutionParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  recipeArn: "RECIPE_ARN" /* required */,
  name: "SOLUTION_NAME" /* required */,
  performAutoTraining: true /* optional, default is true */,
  solutionConfig: {
    autoTrainingConfig: {
      schedulingExpression:
        "rate(5 days)" /* optional, default is every 7 days */,
    },
  },
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSolutionCommand(solutionParam)
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Anda dapat menggunakan kode Python berikut untuk menunggu pelatihan otomatis dimulai. `wait_for_training_to_start`Metode mengembalikan ARN dari versi solusi pertama.

```
import time
import boto3

def wait_for_training_to_start(new_solution_arn):
    max_time = time.time() + 3 * 60 * 60    # 3 hours
    while time.time() < max_time:
        list_solution_versions_response = personalize.list_solution_versions(
            solutionArn=new_solution_arn
        )
        solution_versions = list_solution_versions_response.get('solutionVersions', [])
```

```
        if solution_versions:
            new_solution_version_arn = solution_versions[0]['solutionVersionArn']
            print(f"Solution version ARN: {new_solution_version_arn}")
            return new_solution_version_arn
        else:
            print(f"Training hasn't started yet. Training will start within the next
 hour.")
            time.sleep(60)


personalize = boto3.client('personalize')

solution_arn = "solution_arn"
solution_version_arn = wait_for_training_to_start(solution_arn)
```

Mengkonfigurasi kolom yang digunakan saat pelatihan

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Jika resep Anda menghasilkan rekomendasi item atau segmen pengguna, Anda dapat memodifikasi kolom yang dipertimbangkan Amazon Personalize saat membuat versi solusi (melatih model).

Anda dapat mengubah kolom yang digunakan saat pelatihan untuk mengontrol data yang digunakan Amazon Personalize saat melatih model (membuat versi solusi). Anda dapat melakukan ini untuk bereksperimen dengan kombinasi data pelatihan yang berbeda. Atau Anda mungkin mengecualikan kolom tanpa data yang berarti. Misalnya, mungkin memiliki kolom yang ingin Anda gunakan hanya untuk memfilter rekomendasi. Anda dapat mengecualikan kolom ini dari pelatihan dan Amazon Personalize mempertimbangkannya hanya saat memfilter.

Anda tidak dapat mengecualikan kolom EVENT_TYPE. Secara default, Amazon Personalize menggunakan semua kolom yang dapat digunakan saat pelatihan. Data berikut selalu dikecualikan dari pelatihan:

- Kolom dengan tipe data boolean
- Data tayangan
- Bidang string kustom yang tidak kategoris atau tekstual

Anda tidak dapat menyertakan data tayangan dalam pelatihan, tetapi jika kasus penggunaan atau resep Anda menggunakannya, Amazon Personalize menggunakan data tayangan untuk memandu eksplorasi saat Anda mendapatkan rekomendasi.

Jika Anda telah membuat solusi dan Anda ingin memodifikasi kolom yang digunakannya saat pelatihan, Anda dapat mengkloning solusinya. Saat Anda mengkloning solusi, Anda dapat menggunakan konfigurasi solusi yang ada sebagai titik awal, seperti resep dan hiperparameter, dan membuat perubahan apa pun seperlunya. Untuk informasi selengkapnya, lihat Mengkloning solusi (konsol).

Anda dapat mengonfigurasi kolom yang digunakan Amazon Personalize saat berlatih dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau SDK. AWS Untuk informasi tentang memilih kolom dengan konsol Amazon Personalize, lihat langkah-langkah konfigurasi lanjutan di. Membuat solusi (konsol) Setelah membuat solusi, Anda dapat melihat kolom yang digunakan solusi di halaman detail solusi di konsol Amazon Personalize, atau dengan operasi. DescribeSolution

Topik

- Mengkonfigurasi kolom yang digunakan saat pelatihan ()AWS CLI
- Mengkonfigurasi kolom yang digunakan saat pelatihan ()AWS SDKs

Mengkonfigurasi kolom yang digunakan saat pelatihan ()AWS CLI

Untuk mengecualikan kolom dari pelatihan, berikan `excludedDatasetColumns` objek di `trainingDataConfig` sebagai bagian dari konfigurasi solusi. Untuk setiap kunci, berikan jenis dataset. Untuk setiap nilai, berikan daftar kolom untuk dikecualikan. Kode berikut menunjukkan cara mengecualikan kolom dari pelatihan saat Anda membuat solusi dengan AWS CLI.

```
aws personalize create-solution \
--name solution name \
--dataset-group-arn dataset group ARN \
--recipe-arn recipe ARN \
```

```
--solution-config "{\"trainingDataConfig\": {\"excludedDatasetColumns\":
 { \"datasetType\" : [ \"column1Name\", \"column2Name\"]}}}"
```

Mengkonfigurasi kolom yang digunakan saat pelatihan ()AWS SDKs

Untuk mengecualikan kolom dari pelatihan, berikan `excludedDatasetColumns` objek di `trainingDataConfig` sebagai bagian dari konfigurasi solusi. Untuk setiap kunci, berikan jenis dataset. Untuk setiap nilai, berikan daftar kolom untuk dikecualikan. Kode berikut menunjukkan cara mengecualikan kolom dari pelatihan saat Anda membuat solusi dengan SDK for Python (Boto3).

```
import boto3

personalize = boto3.client('personalize')

create_solution_response = personalize.create_solution(
  name = 'solution name',
  recipeArn = 'recipe ARN',
  datasetGroupArn = 'dataset group ARN',
  solutionConfig = {
    "trainingDataConfig": {
      "excludedDatasetColumns": {
          "datasetType": ["COLUMN_A", "COLUMN_B"]
      }
    }
  }
)
solution_arn = create_solution_response['solutionArn']
print('solution_arn: ', solution_arn)
```

Mengoptimalkan solusi untuk tujuan tambahan

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Jika Anda menggunakan resep Personalisasi Pengguna atau resep Personalisasi Peringkat, Anda dapat mengoptimalkan solusi Amazon Personalize untuk tujuan selain relevansi maksimum, seperti memaksimalkan pendapatan.

Dengan resep rekomendasi item, tujuan utama Amazon Personalize adalah untuk memprediksi item yang paling relevan untuk pengguna Anda berdasarkan data interaksi item historis dan real-time. Ini adalah item yang kemungkinan besar akan berinteraksi dengan pengguna Anda (misalnya, item yang kemungkinan besar akan mereka klik). Jika Anda memiliki tujuan tambahan, seperti memaksimalkan menit streaming atau meningkatkan pendapatan, Anda dapat membuat solusi yang menghasilkan rekomendasi berdasarkan relevansi dan tujuan Anda.

Untuk mengoptimalkan solusi untuk tujuan tambahan, buat solusi baru dengan resep Personalisasi Pengguna atau resep Personalisasi Peringkat dan pilih kolom metadata numerik dalam kumpulan data Item Anda yang terkait dengan tujuan Anda. Saat membuat rekomendasi, Amazon Personalize lebih mementingkan item dengan nilai lebih tinggi untuk kolom data ini. Misalnya, Anda dapat memilih kolom VIDEO_LENGTH untuk memaksimalkan menit streaming atau kolom PRICE untuk memaksimalkan pendapatan.

Anda dapat menggunakan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs Untuk informasi tentang menggunakan konsol Amazon Personalize, lihat. Membuat solusi (konsol)

Topik

- Pedoman dan persyaratan
- Menyeimbangkan penekanan dan relevansi obyektif
- Mengukur kinerja optimasi
- Mengoptimalkan solusi ()AWS CLI
- Mengoptimalkan solusi ()AWS SDKs
- Contoh notebook Jupyter

Pedoman dan persyaratan

Persyaratan obyektif adalah sebagai berikut:

- Anda hanya dapat memilih satu kolom untuk tujuan Anda.
- Kolom harus memiliki tipe numerik dalam skema Anda.
- Kolom tidak dapat memiliki `null` tipe dalam skema Anda.

Untuk informasi selengkapnya tentang skema dan tipe data, lihatMembuat file JSON skema untuk skema Amazon Personalize.

Menyeimbangkan penekanan dan relevansi obyektif

Mungkin ada trade-off ketika merekomendasikan item berdasarkan lebih pada tujuan Anda daripada relevansi. Misalnya, jika Anda ingin meningkatkan pendapatan melalui rekomendasi, rekomendasi hanya untuk item mahal mungkin membuat item kurang relevan bagi pengguna Anda dan mengurangi keterlibatan dan konversi pengguna.

Untuk mengonfigurasi keseimbangan antara relevansi dan tujuan Anda, pilih salah satu dari tingkat sensitivitas objektif berikut saat Anda membuat solusi:

- Mati: Amazon Personalize terutama menggunakan data interaksi item untuk memprediksi item yang paling relevan bagi pengguna Anda.
- Rendah: Amazon Personalize kurang menekankan pada tujuan Anda. Relevansi melalui data interaksi item lebih penting.
- Medium: Amazon Personalize memberikan penekanan yang sama pada tujuan dan relevansi Anda melalui data interaksi item.
- Tinggi: Amazon Personalize lebih menekankan pada tujuan Anda. Relevansi melalui data interaksi item kurang penting.

Mengukur kinerja optimasi

Saat Anda membuat versi solusi (melatih model) untuk solusi dengan tujuan pengoptimalan, Amazon Personalize menghasilkan metrik. `average_rewards_at_k` Skor untuk `average_rewards_at_k` memberi tahu Anda seberapa baik kinerja versi solusi dalam mencapai tujuan Anda. Untuk menghitung metrik ini, Amazon Personalize menghitung hadiah untuk setiap pengguna sebagai berikut:

```
rewards_per_user = total rewards from the user's interactions with their
top 25 reward generating recommendations / total rewards from the user's
interactions with recommendations
```

Final `average_rewards_at_k` adalah rata-rata semua `rewards_per_user` dinormalisasi menjadi nilai desimal kurang dari atau sama dengan 1 dan lebih besar dari 0. Semakin dekat nilainya ke 1, semakin banyak keuntungan rata-rata per pengguna yang dapat Anda harapkan dari rekomendasi.

Misalnya, jika tujuan Anda adalah memaksimalkan pendapatan dari klik, Amazon Personalize menghitung setiap skor pengguna dengan membagi total pendapatan yang dihasilkan oleh item yang diklik pengguna dari 25 rekomendasi termahal teratas mereka dengan pendapatan dari semua item yang direkomendasikan yang diklik pengguna. Amazon Personalize kemudian mengembalikan rata-rata yang dinormalisasi dari semua skor pengguna. Semakin `average_rewards_at_k` dekat ke 1, semakin banyak pendapatan rata-rata yang dapat Anda peroleh per pengguna dari rekomendasi.

Untuk informasi selengkapnya tentang menghasilkan metrik, lihatMengevaluasi versi solusi Amazon Personalize dengan metrik.

Mengoptimalkan solusi ()AWS CLI

Anda dapat mengoptimalkan tujuan hanya dengan resep User-Personalization atau Personalized-Ranking. Untuk mengoptimalkan solusi untuk tujuan tambahan menggunakan AWS CLI, buat solusi baru dan tentukan detail objektif Anda menggunakan `optimizationObjective` kunci dalam `solutionConfig` objek. Ini `optimizationObjective` memiliki bidang-bidang berikut:

- `itemAttribute`: Tentukan nama kolom metadata numerik dari kumpulan data Item yang berhubungan dengan tujuan Anda.
- `objectiveSensitivity`: Tentukan tingkat penekanan yang diberikan solusi pada tujuan Anda saat menghasilkan rekomendasi. Tingkat sensitivitas obyektif mengonfigurasi cara Amazon Personalize menyeimbangkan merekomendasikan item berdasarkan tujuan Anda versus relevansi melalui data data interaksi item. `objectiveSensitivity`Bisa jadi`OFF`, RENDAH, `MEDIUM` atau`HIGH`. Untuk informasi selengkapnya, lihat Menyeimbangkan penekanan dan relevansi obyektif.

Berikut ini adalah contoh perintah `create-solution` AWS CLI . Ganti`solution name`,`dataset group arn`, dan `recipe arn` nilai dengan nilai Anda sendiri.

Untuk`optimizationObjective`, ganti `COLUMN_NAME` dengan nama kolom metadata numerik dari kumpulan data Item yang terkait dengan tujuan Anda. Untuk`objectiveSensitivity`, tentukan OFF, LOW, MEDIUM, atau HIGH.

```
aws personalize create-solution \
--name solution name \
--dataset-group-arn dataset group arn \
--recipe-arn recipe arn \
--solution-config "{\"optimizationObjective\":{\"itemAttribute\":\"COLUMN_NAME\",
\"objectiveSensitivity\":\"MEDIUM\"}}"
```

Saat solusi Anda siap, buat versi solusi baru (untuk contoh perintah lihatMembuat solusi (AWS CLI)). Setelah membuat versi solusi, Anda dapat melihat kinerja pengoptimalan dengan metrik versi solusi. Lihat Mengukur kinerja optimasi.

Mengoptimalkan solusi ()AWS SDKs

Anda dapat mengoptimalkan tujuan hanya dengan resep User-Personalization atau Personalized-Ranking.

Untuk mengoptimalkan solusi untuk tujuan tambahan menggunakan AWS SDKs, buat solusi baru dan tentukan detail objektif Anda menggunakan `optimizationObjective` kunci dalam `solutionConfig` objek untuk solusi tersebut. Ini `optimizationObjective` memiliki bidang-bidang berikut:

- `itemAttribute`: Tentukan nama kolom metadata numerik dari kumpulan data Kumpulan data Item yang terkait dengan tujuan Anda.
- `objectiveSensitivity`: Tentukan tingkat penekanan yang diberikan solusi pada tujuan Anda saat menghasilkan rekomendasi. Tingkat sensitivitas obyektif mengonfigurasi cara Amazon Personalize menyeimbangkan merekomendasikan item berdasarkan tujuan Anda versus relevansi melalui data data interaksi item. `objectiveSensitivity`Bisa jadi`OFF`,`LOW`, `MEDIUM` atau`HIGH`. Untuk informasi selengkapnya, lihat Menyeimbangkan penekanan dan relevansi obyektif.

Gunakan kode berikut untuk membuat solusi dengan tujuan tambahan dengan AWS SDK untuk Python (Boto3) atau AWS SDK for Java 2.x.

Ketika solusi Anda siap, buat versi solusi baru (misalnya kode lihatMembuat versi solusi (AWS SDKs)). Setelah membuat versi solusi, Anda dapat melihat kinerja pengoptimalan dengan metrik versi solusi. Lihat Mengukur kinerja optimasi.

SDK for Python (Boto3)

Untuk membuat solusi yang dioptimalkan untuk tujuan tambahan, gunakan `create_solution` metode berikut. Ganti`solution name`,`dataset group arn`, dan `recipe arn` nilai dengan nilai Anda sendiri.

Untuk`optimizationObjective`, ganti `COLUMN_NAME` dengan nama kolom metadata numerik dari kumpulan data Item yang terkait dengan tujuan Anda. Untuk`objectiveSensitivity`, tentukan OFF, LOW, MEDIUM, atau HIGH.

```
import boto3

personalize = boto3.client('personalize')

create_solution_response = personalize.create_solution(
    name= 'solution name',
    recipeArn = 'recipe arn',
    datasetGroupArn = 'dataset group arn',
    solutionConfig = {
        "optimizationObjective": {
            "itemAttribute": "COLUMN_NAME",
            "objectiveSensitivity": "MEDIUM"
        }
    }
)
solution_arn = create_solution_response['solutionArn']
print('solution_arn: ', solution_arn)
```

SDK for Java 2.x

Untuk membuat solusi yang dioptimalkan untuk tujuan tambahan, gunakan `createPersonalizeSolution` metode berikut dan teruskan yang berikut ini sebagai parameter: klien layanan Amazon Personalize, Amazon Resource Name (ARN) grup dataset, nama solusi, ARN resep, atribut item, dan tingkat sensitivitas objektif.

```
public static String createPersonalizeSolution(PersonalizeClient personalizeClient,
                                                String datasetGroupArn,
                                                String solutionName,
                                                String recipeArn,
                                                String itemAttribute,
                                                String objectiveSensitivity) {

    try {
        OptimizationObjective optimizationObjective =
 OptimizationObjective.builder()
            .itemAttribute(itemAttribute)
            .objectiveSensitivity(objectiveSensitivity)
            .build();

        SolutionConfig solutionConfig = SolutionConfig.builder()
            .optimizationObjective(optimizationObjective)
            .build();
```

```
        CreateSolutionRequest solutionRequest = CreateSolutionRequest.builder()
            .name(solutionName)
            .datasetGroupArn(datasetGroupArn)
            .recipeArn(recipeArn)
            .solutionConfig(solutionConfig)
            .build();

        CreateSolutionResponse solutionResponse =
 personalizeClient.createSolution(solutionRequest);

        return solutionResponse.solutionArn();

    } catch (PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return "";
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { CreateSolutionCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create the personalizeClient
const personalizeClient = new PersonalizeClient({ region: "REGION"});

// set the solution parameters.
export const createSolutionParam = {
  datasetGroupArn: 'DATASET_GROUP_ARN',             /* required */
  recipeArn: 'RECIPE_ARN',                          /* required */
  name: 'NAME',                                     /* required */
  solutionConfig: {
    optimizationObjective: {
      itemAttribute: "COLUMN_NAME",           /* specify the numerical column from
 the Items dataset related to your objective */
      objectiveSensitivity: "MEDIUM"          /* specify OFF, LOW, MEDIUM, or HIGH
 */
    }
  }
};
```

```
export const run = async () => {
  try {
    const response = await personalizeClient.send(new
 CreateSolutionCommand(createSolutionParam));
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Contoh notebook Jupyter

Untuk contoh notebook Jupyter yang menunjukkan cara membuat solusi yang dioptimalkan untuk metadata item berbasis objektif tambahan, lihat folder objective_optimization dari repositori Amazon Personalize samples GitHub

Mengoptimalkan solusi dengan konfigurasi peristiwa

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Jika Anda menggunakan resep User-Personalization-v 2 atau Personalized-Ranking-v 2 resep, Anda dapat mengoptimalkan solusi Amazon Personalize dengan konfigurasi peristiwa.

Dengan resep rekomendasi item, tujuan utama Amazon Personalize untuk memprediksi item yang paling relevan bagi pengguna Anda berdasarkan data interaksi item historis dan real-time. Namun, interaksi dapat membawa informasi tambahan seperti apakah pengguna mengklik atau membeli item tertentu. Anda dapat merekam ini dengan merekam jenis acara (Jenis peristiwa dan data nilai acara). Ketika Anda mengonfigurasi solusi, Anda dapat meminta solusi memberikan bobot yang berbeda untuk jenis peristiwa interaksi yang berbeda. Misalnya, Anda dapat mengonfigurasi solusi untuk memberi bobot lebih pada `purchase` peristiwa daripada `click` acara.

Agar solusi memberikan bobot yang berbeda untuk jenis peristiwa yang berbeda, Anda menentukan jenis acara dan bobot yang sesuai dalam konfigurasi acara solusi. Selain itu, Anda dapat menetapkan ambang nilai peristiwa untuk mengecualikan interaksi dengan nilai peristiwa di bawah ambang batas tersebut. Misalnya, jika data EVENT_VALUE untuk peristiwa dengan EVENT_TYPE jam tangan adalah persentase video yang ditonton pengguna, jika Anda menetapkan ambang nilai peristiwa ke 0,5, dan jenis acara yang akan ditonton, Amazon Personalize melatih model hanya menggunakan menonton peristiwa interaksi dengan EVENT_VALUE lebih besar dari atau sama dengan 0,5.

Bobot yang terkait dengan jenis acara akan menentukan kepentingannya. Jenis acara dengan bobot lebih tinggi akan menyebabkan model terlatih lebih mungkin merekomendasikan item yang akan berinteraksi dengan jenis acara tersebut. Misalnya, jika Anda menentukan “pembelian” dengan bobot lebih tinggi dari “klik” dan model mengetahui bahwa pengguna akan mengklik item B atau membeli item C dengan riwayat interaksinya, model akan memberi peringkat item C lebih tinggi.

Untuk mengoptimalkan solusi dengan konfigurasi peristiwa, Anda membuat solusi baru dengan resep User-Personalization-v 2 atau Personalized-Ranking-v 2 resep dan menentukan konfigurasi peristiwa. Anda juga dapat memperbarui solusi (Memperbarui solusi untuk mengubah konfigurasi pelatihan otomatisnya) yang ada dengan konfigurasi peristiwa.

Anda dapat menggunakan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs Untuk informasi tentang menggunakan konsol Amazon Personalize, lihat. Membuat solusi (konsol)

Topik

- Pedoman dan persyaratan
- Mengukur kinerja dengan konfigurasi bobot acara
- Mengoptimalkan solusi ()AWS CLI
- Mengoptimalkan solusi ()AWS SDKs

Pedoman dan persyaratan

Berikut ini adalah pedoman dan persyaratan untuk konfigurasi acara:

- Untuk mengonfigurasi bobot untuk jenis peristiwa yang berbeda, kumpulan data kumpulan data interaksi Item Anda harus memiliki kolom EVENT_TYPE dan secara opsional kolom EVENT_VALUE.

- Anda dapat menentukan daftar parameter acara dalam konfigurasi. Sertakan semua jenis acara yang ingin Anda pertimbangkan untuk pembuatan solusi. Anda dapat menentukan maksimal 10 jenis acara yang berbeda.
- Anda dapat menentukan bobot acara untuk setiap jenis acara. Berat acara harus antara 0,0 dan 1,0. Hanya rasio bobot antara jenis peristiwa yang penting. Misalnya, menyetel jenis acara “pembelian” dengan berat 0,3 dan jenis acara “klik” dengan berat 0,1 akan memiliki efek yang sama dengan pengaturan “pembelian dengan berat 0,6 dan” klik “dengan berat 0,2.
- Anda dapat memperbarui konfigurasi acara untuk solusi yang ada menggunakan operasi UpdateSolutionAPI.

Mengukur kinerja dengan konfigurasi bobot acara

Saat Anda membuat versi solusi (melatih model) untuk solusi dengan konfigurasi peristiwa, Amazon Personalize akan menghasilkan metrik. `normalized_discounted_cumulative_gain_with_event_weights_at_k` Skor untuk `normalized_discounted_cumulative_gain_with_event_weights_at_k` memberi tahu Anda seberapa baik kinerja versi solusi mengingat bobot acara yang Anda tetapkan untuk setiap jenis acara.

Ini mirip dengan keuntungan kumulatif diskon yang dinormalisasi (NDCG) di K tetapi hadiah untuk setiap prediksi yang benar akan tertimbang. Sebaliknya, dalam NDCG asli di K, setiap prediksi yang benar semuanya akan membawa bobot 1. Misalnya, dengan “pembelian” berat 0,3 dan “klik” berat 0,1, memprediksi dengan benar item “pembelian” akan mendapatkan hadiah 1,5 sementara memprediksi item “klik” akan mendapatkan hadiah 0,5.

Untuk informasi selengkapnya tentang menghasilkan metrik, lihatMengevaluasi versi solusi Amazon Personalize dengan metrik.

Mengoptimalkan solusi ()AWS CLI

Anda dapat mengoptimalkan dengan konfigurasi acara dengan resep User-Personalization-v Personalized-Ranking-v 2 atau 2.

Untuk mengoptimalkan solusi dengan konfigurasi peristiwa menggunakan AWS CLI, buat solusi baru dan tentukan detail konfigurasi acara Anda menggunakan `eventsConfig` kunci di `solutionConfig` objek. Ini `eventsConfig` memiliki kunci `eventParametersList` di mana Anda dapat menentukan hingga 10 EventParameters. Masing-masing `eventParameter` memiliki bidang berikut:

- EventType: tentukan jenis acara yang ingin Anda pertimbangkan untuk pembuatan solusi.
- eventValueThreshold: tentukan ambang nilai acara. Hanya peristiwa dengan nilai peristiwa yang lebih besar atau sama dengan ambang batas ini yang akan dipertimbangkan untuk pembuatan solusi.
- berat: tentukan bobot untuk setiap jenis acara. Bobot yang lebih tinggi berarti lebih penting dari jenis acara untuk solusi yang dibuat.

Berikut ini adalah contoh dari perintah create-solution AWS CLI . Ganti`solution name`,`dataset group arn`, dan `recipe arn` nilai dengan nilai Anda sendiri.

```
aws personalize create-solution \
--name solution name \
--dataset-group-arn dataset group arn \
--recipe-arn recipe arn \
--solution-config "{\"eventsConfig\":{\"eventParametersList\":[{\"eventType\":
\"Purchase\", \"eventValueThreshold\":0.1, \"weight\":0.3}, {\"eventType\":\"Click\",
 \"weight\":0.1}]}"
```

Saat solusi Anda siap, buat versi solusi baru (untuk contoh perintah lihatMembuat solusi (AWS CLI)). Setelah membuat versi solusi, Anda dapat melihat kinerja pengoptimalan dengan metrik versi solusi. Lihat Mengukur kinerja optimasi.

Mengoptimalkan solusi ()AWS SDKs

Anda dapat mengoptimalkan dengan konfigurasi acara dengan resep User-Personalization-v Personalized-Ranking-v 2 atau 2.

Untuk mengoptimalkan solusi dengan konfigurasi peristiwa menggunakan AWS SDKs, buat solusi baru dan tentukan detail konfigurasi acara Anda menggunakan `eventsConfig` kunci di `solutionConfig` objek. Ini `eventsConfig` memiliki kunci `eventParametersList` di mana Anda dapat menentukan hingga 10`eventParameters`. Masing-masing `eventParameter` memiliki bidang berikut:

- EventType: tentukan jenis acara yang ingin Anda pertimbangkan untuk pembuatan solusi.
- eventValueThreshold: tentukan ambang nilai acara. Hanya peristiwa dengan nilai peristiwa yang lebih besar atau sama dengan ambang batas ini yang akan dipertimbangkan untuk pembuatan solusi.

- berat: tentukan bobot untuk setiap jenis acara. Bobot yang lebih tinggi berarti lebih penting dari jenis acara untuk solusi yang dibuat.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

create_solution_response = personalize.create_solution(
    name= 'solution name',
    recipeArn = 'recipe arn',
    datasetGroupArn = 'dataset group arn',
    solutionConfig = {
       "eventsConfig": {
          "eventParametersList": [
             {"eventType":"Purchase",
              "eventValueThreshold":0.1,
              "weight":0.3},
             {"eventType":"Click",
              "weight":0.1}
          ]
       }
    }
)
solution_arn = create_solution_response['solutionArn']
print('solution_arn: ', solution_arn)
```

SDK for Java 2.x

```
public static String createPersonalizeSolution(PersonalizeClient personalizeClient,
         String datasetGroupArn,
         String solutionName,
         String recipeArn,
         ) {

try {
    EventsConfig eventsConfig = EventsConfig.builder()
        .eventsParameterList(eventsParameterList)
        .build();

    SolutionConfig solutionConfig = SolutionConfig.builder()
```

```
            .eventsConfig(eventsConfig)
            .build();

        CreateSolutionRequest solutionRequest = CreateSolutionRequest.builder()
            .name(solutionName)
            .datasetGroupArn(datasetGroupArn)
            .recipeArn(recipeArn)
            .solutionConfig(solutionConfig)
            .build();

        CreateSolutionResponse solutionResponse =
     personalizeClient.createSolution(solutionRequest);

        return solutionResponse.solutionArn();

    } catch (PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return "";
```

Saat solusi Anda siap, buat versi solusi baru (untuk contoh perintah lihatMembuat solusi (AWS SDKs)). Setelah membuat versi solusi, Anda dapat melihat kinerja pengoptimalan dengan metrik versi solusi. Lihat Mengukur kinerja optimasi.

Hyperparameters dan HPO

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Anda menentukan hyperparameters sebelum pelatihan untuk mengoptimalkan model terlatih untuk kasus penggunaan khusus Anda. Ini kontras dengan parameter model yang nilainya ditentukan selama proses pelatihan.

Hyperparameters ditentukan menggunakan `algorithmHyperParameters` kunci yang merupakan bagian dari SolutionConfig objek yang diteruskan ke CreateSolution operasi.

Versi `CreateSolution` permintaan yang diringkas ada di bawah ini. Contohnya termasuk `solutionConfig` objek. Anda gunakan `solutionConfig` untuk mengganti parameter default resep.

```
{
  "name": "string",
  "recipeArn": "string",
  "eventType": "string",
  "solutionConfig": {
      "optimizationObjective": {
          "itemAttribute": "string",
          "objectiveSensitivity": "string"
      },
      "eventValueThreshold": "string",
      "featureTransformationParameters": {
          "string" : "string"
      },
      "algorithmHyperParameters": {
          "string" : "string"
      },
      "hpoConfig": {
          "algorithmHyperParameterRanges": {
              ...
          },
          "hpoResourceConfig": {
              "maxNumberOfTrainingJobs": "string",
              "maxParallelTrainingJobs": "string"
          }
      },
  },
}
```

Resep yang berbeda menggunakan hyperparameters yang berbeda. Untuk hiperparameter yang tersedia, lihat resep individual diMemilih resep.

Mengaktifkan optimasi hyperparameter

Optimasi hiperparameter (HPO), atau penyetelan, adalah tugas memilih hiperparameter optimal untuk tujuan pembelajaran tertentu. Hiperparameter optimal ditentukan dengan menjalankan banyak pekerjaan pelatihan menggunakan nilai yang berbeda dari rentang kemungkinan yang ditentukan.

Dengan User-personalization-v2 dan Personalized-Ranking-v2, jika Anda mengaktifkan pelatihan otomatis, Amazon Personalize secara otomatis melakukan HPO setiap 90 hari. Tanpa pelatihan otomatis, tidak ada HPO yang terjadi. Untuk semua resep lainnya, Anda harus mengaktifkan HPO. Untuk menggunakan HPO, atur `performHPO` ke `true`, dan sertakan `hpoConfig` objek.

Hyperparameter dapat berupa kategoris, kontinu, atau bernilai bilangan bulat. `hpoConfig`Objek memiliki kunci yang sesuai dengan masing-masing jenis ini, di mana Anda menentukan hyperparameters dan rentangnya. Anda harus memberikan setiap jenis dalam permintaan Anda, tetapi jika resep tidak memiliki parameter tipe, Anda dapat membiarkannya kosong. Misalnya, User-Personalization tidak memiliki hyperparameter yang dapat disetel dari tipe kontinu. Jadi untuk`continousHyperParameterRange`, Anda akan melewati array kosong.

Kode berikut menunjukkan cara membuat solusi dengan HPO diaktifkan menggunakan SDK for Python (Boto3). Solusi dalam contoh menggunakan Resep Personalisasi Pengguna resep dan memiliki HPO diatur `ke true`. Kode memberikan nilai untuk `hidden_dimension` dan `categoricalHyperParameterRanges` dan`integerHyperParameterRanges`. `continousHyperParameterRange`Yang kosong dan `hpoResourceConfig` set `maxNumberOfTrainingJobs` dan`maxParallelTrainingJobs`.

```
import boto3

personalize = boto3.client('personalize')

create_solution_response = personalize.create_solution(
    name = "solution name",
    datasetGroupArn = 'arn:aws:personalize:region:accountId:dataset-group/
datasetGroupName',
    recipeArn = 'arn:aws:personalize:::recipe/aws-user-personalization',
    performHPO = True,
    solutionConfig = {
        "algorithmHyperParameters": {
          "hidden_dimension": "55"
        },
        "hpoConfig": {
```

```
        "algorithmHyperParameterRanges": {
            "categoricalHyperParameterRanges": [
                {
                    "name": "recency_mask",
                    "values": [ "true", "false"]
                }
            ],
            "integerHyperParameterRanges": [
                {
                    "name": "bptt",
                    "minValue": 2,
                    "maxValue": 22
                }
            ],
            "continuousHyperParameterRanges": [

            ]
        },
        "hpoResourceConfig": {
            "maxNumberOfTrainingJobs": "4",
            "maxParallelTrainingJobs": "2"
        }
    }
}
)
```

Untuk informasi selengkapnya tentang HPO, lihat Penyetelan model otomatis.

Melihat hiperparameter

Anda dapat melihat hyperparameters solusi dengan memanggil DescribeSolution operasi. Contoh berikut menunjukkan `DescribeSolution` output. Setelah membuat versi solusi (melatih model), Anda juga dapat melihat hyperparameters dengan DescribeSolutionVersion operasi.

```
{
  "solution": {
    "name": "hpo_coonfig_solution",
    "solutionArn": "arn:aws:personalize:region:accountId:solution/solutionName",
    "performHPO": true,
    "performAutoML": false,
    "recipeArn": "arn:aws:personalize:::recipe/aws-user-personalization",
    "datasetGroupArn": "arn:aws:personalize:region:accountId:dataset-group/
datasetGroupName",
```

```
    "eventType": "click",
    "solutionConfig": {
      "hpoConfig": {
        "hpoResourceConfig": {
          "maxNumberOfTrainingJobs": "4",
          "maxParallelTrainingJobs": "2"
        },
        "algorithmHyperParameterRanges": {
          "integerHyperParameterRanges": [
            {
              "name": "training.bptt",
              "minValue": 2,
              "maxValue": 22
            }
          ],
          "continuousHyperParameterRanges": [],
          "categoricalHyperParameterRanges": [
            {
              "name": "data.recency_mask",
              "values": [
                "true",
                "false"
              ]
            }
          ]
        }
      },
      "algorithmHyperParameters": {
        "hidden_dimension": "55"
      }
    },
    "status": "ACTIVE",
    "creationDateTime": "2022-07-08T12:12:48.565000-07:00",
    "lastUpdatedDateTime": "2022-07-08T12:12:48.565000-07:00"
  }
}
```

Memilih data interaksi item yang digunakan untuk pelatihan

Important

Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari

biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Anda dapat memilih peristiwa dalam kumpulan data interaksi Item yang digunakan Amazon Personalize saat membuat versi solusi (melatih model). Memilih data interaksi item sebelum pelatihan memungkinkan Anda menggunakan hanya subset data yang relevan untuk pelatihan atau menghilangkan noise untuk melatih model yang lebih optimal. Untuk informasi selengkapnya tentang kumpulan data interaksi Item, lihat. Data interaksi item

> **Note**
>
> Jika Anda menggunakan User-Personalization-v 2 atau Personalized-Ranking-v 2, biaya pelatihan Anda didasarkan pada data interaksi item Anda sebelum memfilter berdasarkan jenis atau nilai acara. Untuk informasi selengkapnya tentang harga, lihat Harga Amazon Personalisasi.

Anda dapat memilih data interaksi item sebagai berikut:

- Pilih catatan berdasarkan jenis — Saat Anda mengonfigurasi solusi, jika kumpulan data interaksi Item menyertakan tipe peristiwa di kolom EVENT_TYPE, Anda dapat menentukan jenis peristiwa yang akan digunakan dalam pelatihan secara opsional. Misalnya, jika kumpulan data interaksi Item menyertakan jenis peristiwa pembelian, klik, dan tonton, dan Anda ingin Amazon Personalize melatih model hanya dengan acara tontonan, saat mengonfigurasi solusi, Anda akan memberikan jam tangan sebagai yang digunakan Amazon `event type` Personalize dalam pelatihan.

  Jika Anda memiliki beberapa jenis acara dan menggunakan resep User-Personalization-v 2 atau resep Personalized-Ranking-v 2, saat Anda mengonfigurasi solusi khusus, Anda dapat menentukan bobot yang berbeda untuk jenis yang berbeda. Misalnya, Anda dapat mengonfigurasi solusi untuk memberikan bobot lebih untuk membeli acara daripada peristiwa klik. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi dengan konfigurasi peristiwa.

  Jika kumpulan data interaksi Item memiliki beberapa tipe peristiwa di kolom EVENT_TYPE, dan Anda tidak memberikan jenis peristiwa saat mengonfigurasi solusi, Amazon Personalize menggunakan semua data interaksi item untuk pelatihan dengan bobot yang sama terlepas dari jenisnya.

- Pilih catatan berdasarkan jenis dan nilai — Saat Anda mengonfigurasi solusi, jika kumpulan data interaksi Item menyertakan bidang EVENT_TYPE dan EVENT_VALUE, Anda dapat menetapkan nilai tertentu sebagai ambang batas untuk mengecualikan catatan dari pelatihan. Misalnya, jika data EVENT_VALUE untuk peristiwa dengan EVENT_TYPE jam tangan adalah persentase video yang ditonton pengguna, jika Anda menetapkan ambang nilai peristiwa ke 0,5, dan jenis acara yang akan ditonton, Amazon Personalize melatih model hanya menggunakan menonton peristiwa interaksi dengan EVENT_VALUE lebih besar dari atau sama dengan 0,5.

Kode berikut menunjukkan cara menggunakan SDK for Python (Boto3) untuk membuat solusi yang hanya `watch` menggunakan peristiwa di mana penggunaan menonton lebih dari setengah video.

```
import boto3

personalize = boto3.client('personalize')

create_solution_response = personalize.create_solution(
    name = 'solution name',
    datasetGroupArn = 'arn:aws:personalize:region:accountId:dataset-group/
datasetGroupName',
    recipeArn = 'arn:aws:personalize:::recipe/aws-user-personalization-v2',
    eventType = 'watch',
    solutionConfig = {
        "eventValueThreshold": "0.5"
    }
)

# Store the solution ARN
solution_arn = create_solution_response['solutionArn']

# Use the solution ARN to get the solution status
solution_description = personalize.describe_solution(solutionArn = solution_arn)
['solution']
print('Solution status: ' + solution_description['status'])
```

Mengkloning solusi (konsol)

Saat membuat solusi baru, Anda dapat menggunakan konsol Amazon Personalize untuk mengkloning solusi. Saat Anda mengkloning solusi, Anda dapat menggunakan konfigurasi solusi yang ada sebagai titik awal, seperti resep dan hiperparameter, dan membuat perubahan apa pun seperlunya. Ini berguna jika Anda ingin membuat satu perubahan pada solusi, tetapi biarkan semua

properti lainnya tidak berubah. Misalnya, menambahkan kolom baru data pelatihan ke kumpulan data Anda. Dalam hal ini, Anda akan mengkloning solusi, memberi nama solusi, mengubah kolom yang digunakan saat pelatihan, dan membiarkan semua properti lainnya tidak berubah.

Mengkloning solusi

Untuk mengkloning solusi, Anda memilih solusi yang ada, dan memilih opsi solusi Klon. Kemudian beri nama solusi baru, dan modifikasi bidang yang relevan.

Untuk mengkloning solusi

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
3. Pilih Sumber daya khusus dan pilih Solusi.
4. Pilih solusi yang ingin Anda kloning.
5. Pilih Actions, dan pilih Clone solution.
6. Beri nama solusi baru.
7. Buat perubahan apa pun pada detail solusi dan konfigurasi lanjutan. Amazon Personalize mengisi bidang ini sebelumnya dengan nilai dari solusi yang ada. Untuk informasi tentang setiap bidang, lihatMengonfigurasi solusi khusus di Amazon Personalize.

Memperbarui solusi untuk mengubah konfigurasi pelatihan otomatisnya

Setelah Anda membuat solusi, Anda dapat mengubah konfigurasi pelatihan otomatis dan konfigurasi acaranya:

- Anda dapat mengaktifkan atau menonaktifkan pelatihan otomatis, dan Anda dapat mengubah frekuensi pelatihan.
  - Jika Anda mengaktifkan pelatihan otomatis, pelatihan otomatis pertama dimulai dalam waktu satu jam setelah pembaruan solusi selesai. Jika Anda membuat versi solusi secara manual dalam satu jam, solusinya melewatkan pelatihan otomatis pertama.
  - Jika Anda memodifikasi frekuensi pelatihan solusi, jadwal pelatihan akan diatur ulang dan versi solusi baru memulai pelatihan dalam satu jam. Pembuatan versi solusi berlanjut pada frekuensi baru, di mana hari 1 adalah hari Anda memperbarui solusi.

- Anda dapat memperbarui konfigurasi acara solusi. Jika solusi sudah memiliki konfigurasi peristiwa yang ditentukan, konfigurasi peristiwa baru akan menggantikan yang asli. Untuk informasi tentang mengoptimalkan solusi untuk konfigurasi acara, lihatMengoptimalkan solusi dengan konfigurasi peristiwa.

Anda dapat memperbarui solusi dengan konsol Amazon Personalize, AWS Command Line Interface, atau. AWS SDKs Pembaruan solusi dapat memakan waktu beberapa menit. Saat pembaruan sedang berlangsung, Anda dapat membuat versi solusi untuk solusi tersebut tetapi Anda tidak dapat menghapus solusinya. Sampai pembaruan selesai, solusinya menggunakan konfigurasi sebelumnya. Untuk informasi selengkapnya tentang pelatihan otomatis, lihatMengkonfigurasi pelatihan otomatis.

Topik

- Memperbarui solusi (konsol)
- Memperbarui solusi (AWS CLI)
- Memperbarui solusi (AWS SDKs)

Memperbarui solusi (konsol)

Untuk memperbarui solusi di konsol, navigasikan ke solusi, pilih perbarui, dan tentukan konfigurasi baru yang akan digunakan.

Untuk mengkonfigurasi solusi

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah, dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
3. Di panel navigasi, pilih Sumber daya khusus dan pilih Solusi dan resep.
4. Pilih solusi Anda dan pilih Perbarui di kanan atas.
5. Dalam pelatihan otomatis, modifikasi apakah solusi menggunakan pelatihan otomatis. Jika pelatihan otomatis aktif, Anda dapat mengubah`Automatic training frequency`. Frekuensi pelatihan default adalah setiap 7 hari.
6. Pilih solusi Perbarui. Anda dapat menemukan status pembaruan solusi di halaman detail solusi Anda.

Memperbarui solusi (AWS CLI)

Untuk memperbarui solusi dengan AWS Command Line Interface, gunakan `update-solution` perintah. Perintah ini menggunakan operasi UpdateSolution API. Kode berikut menunjukkan kepada Anda cara memperbarui solusi untuk menggunakan pelatihan otomatis dengan frekuensi pelatihan 5 hari. Untuk mematikan pelatihan otomatis, tentukan `--no-perform-auto-training` dan hilangkan. `solution-update-config`

Frekuensi pelatihan default adalah setiap 7 hari. Ekspresi harus dalam `rate(value unit)` format. Untuk nilainya, tentukan angka antara 1 dan 30. Untuk unit, tentukan `day` atau`days`.

```
aws personalize update-solution \
--solution-arn solution ARN \
--perform-auto-training \
--solution-update-config "{\"autoTrainingConfig\": {\"schedulingExpression\": \"rate(5
 days)\"}}"
```

Untuk mendapatkan status pembaruan, gunakan `describe-solution` perintah (yang menggunakan operasi DescribeSolution API) dan temukan status pembaruan di`latestSolutionUpdate`.

Memperbarui solusi (AWS SDKs)

Untuk memperbarui solusi dengan AWS SDKs, gunakan operasi UpdateSolution API. Kode berikut menunjukkan kepada Anda cara menggunakan SDK for Python (Boto3) untuk memperbarui solusi untuk menggunakan pelatihan otomatis dengan frekuensi pelatihan 5 hari. Kode mendapatkan status pembaruan dengan operasi DescribeSolution API.

Frekuensi pelatihan default adalah setiap 7 hari. Ekspresi harus dalam `rate(value unit)` format. Untuk nilainya, tentukan angka antara 1 dan 30. Untuk unit, tentukan `day` atau`days`.

```
import boto3

personalize = boto3.client('personalize')

update_solution_response = personalize.update_solution(
    solutionArn='SOLUTION ARN',
    performAutoTraining=True,
    solutionUpdateConfig={
        "autoTrainingConfig": {
            "schedulingExpression": "rate(5 days)"
```

```
        }
    }
)
describe_solution_response = personalize.describe_solution(
    solutionArn='SOLUTION ARN'
)
update_status = describe_solution_response["solution"]["latestSolutionUpdate"]
["status"]
print(f"Update status: {update_status}")
```

Membuat versi solusi secara manual

Setelah selesaiMengonfigurasi solusi khusus di Amazon Personalize, Anda siap untuk memulai pelatihan:

- Jika solusi Anda menggunakan pelatihan otomatis, solusi akan membuat versi solusi untuk Anda pada frekuensi pelatihan yang Anda tentukan. Secara default, semua solusi baru menggunakan pelatihan otomatis untuk membuat versi solusi baru setiap 7 hari. Anda masih dapat membuat versi solusi secara manual. Untuk informasi selengkapnya, lihat Mengkonfigurasi pelatihan otomatis.
- Jika Anda mematikan pelatihan otomatis untuk solusi Anda atau Anda ingin melatih secara manual, Anda dapat membuat versi solusi secara manual. Versi solusi mengacu pada model pembelajaran mesin yang terlatih. Anda dapat membuat versi solusi menggunakan konsol, AWS Command Line Interface (AWS CLI), atau AWS SDKs. Jika versi solusi Anda memiliki status CREATE_PENDING atau CREATE_IN_PROGRESS, Anda dapat menggunakan the section called "StopSolutionVersionCreation" operasi untuk menghentikan proses pembuatan versi solusi. Lihat Menghentikan pembuatan versi solusi.

Jika pelatihan tidak selesai karena kesalahan, Anda tidak dikenakan biaya untuk pelatihan. Jika versi solusi Anda memiliki status CREATE_PENDING atau CREATE_IN_PROGRESS, Anda dapat menghentikan proses pembuatan versi solusi. Untuk menghentikan pembuatan versi solusi, navigasikan ke halaman detail versi solusi dan pilih Berhenti. Untuk informasi selengkapnya, lihat Menghentikan pembuatan versi solusi.

Topik

- Membuat versi solusi (konsol)
- Membuat versi solusi (AWS CLI)
- Membuat versi solusi (AWS SDKs)

Membuat versi solusi (konsol)

Untuk membuat versi solusi baru secara manual dengan konsol Amazon Personalize, Anda memulai pelatihan dari halaman detail solusi Anda.

Untuk membuat versi solusi baru

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Arahkan ke halaman grup kumpulan data dan pilih grup kumpulan data dengan solusi baru Anda.
3. Di panel navigasi, di bawah Sumber daya khusus, pilih Solusi dan resep.
4. Pada halaman Solusi dan resep, pilih solusi yang ingin Anda buat versi solusinya.
5. Pada halaman ikhtisar solusi, pilih Buat versi solusi untuk mulai melatih model baru.

Pada halaman detail solusi, Anda dapat melacak kemajuan pelatihan di bagian Versi solusi. Saat pelatihan selesai, statusnya Aktif, Anda dapat mengevaluasinya menggunakan metrik yang disediakan oleh Amazon Personalize. Untuk informasi selengkapnya, lihat Mengevaluasi versi solusi Amazon Personalize dengan metrik.

Ketika versi solusi AKTIF, Anda siap menggunakannya untuk mendapatkan rekomendasi. Bagaimana Anda menggunakan versi solusi aktif tergantung pada bagaimana Anda mendapatkan rekomendasi:

- Untuk rekomendasi real-time, Anda menerapkan versi solusi AKTIF dengan kampanye Amazon Personalize. Anda menggunakan kampanye untuk mendapatkan rekomendasi bagi pengguna Anda. Lihat Menerapkan versi solusi Amazon Personalize dengan kampanye.
- Untuk rekomendasi batch, Anda menentukan versi solusi AKTIF saat membuat pekerjaan inferensi batch atau pekerjaan segmen batch. Lihat Mendapatkan rekomendasi item batch atau Mendapatkan segmen pengguna batch.

Membuat versi solusi (AWS CLI)

Ketika solusi Anda AKTIF, latih model dengan menjalankan perintah berikut. Ganti `solution arn` dengan solusi Amazon Resource Name (ARN) dari. Mengonfigurasi solusi khusus di Amazon Personalize

```
aws personalize create-solution-version \
```

```
  --solution-arn solution arn
```

Versi solusi ARN ditampilkan, misalnya:

```
{
  "solutionVersionArn": "arn:aws:personalize:us-west-2:acct-id:solution/SolutionName/
<version-id>"
}
```

Periksa status pelatihan versi solusi dengan menggunakan `describe-solution-version` perintah. Berikan solusi versi ARN yang dikembalikan pada langkah sebelumnya. Untuk informasi selengkapnya tentang API, lihat DescribeSolutionVersion.

```
aws personalize describe-solution-version \
  --solution-version-arn solution version arn
```

Properti versi solusi dan pelatihan `status` ditampilkan. Awalnya, status ditampilkan sebagai CREATE PENDING, misalnya:

```
{
  "solutionVersion": {
      "solutionVersionArn": "arn:aws:personalize:us-west-2:acct-id:solution/
solutionName/<version-id>",
      ...,
      "status": "CREATE PENDING"
  }
}
```

Pelatihan selesai saat `status` selesai `ACTIVE` dan Anda dapat mengevaluasinya menggunakan metrik yang disediakan oleh Amazon Personalize. Untuk informasi selengkapnya, lihat Mengevaluasi versi solusi Amazon Personalize dengan metrik. Jika pelatihan tidak selesai karena kesalahan, Anda tidak dikenakan biaya untuk pelatihan.

Jika versi solusi Anda memiliki status CREATE_PENDING atau CREATE_IN_PROGRESS, Anda dapat menggunakan StopSolutionVersionCreation operasi untuk menghentikan proses pembuatan versi solusi. Lihat Menghentikan pembuatan versi solusi.

Ketika versi solusi AKTIF, Anda siap menggunakannya untuk mendapatkan rekomendasi. Bagaimana Anda menggunakan versi solusi aktif tergantung pada bagaimana Anda mendapatkan rekomendasi:

- Untuk rekomendasi real-time, Anda menerapkan versi solusi AKTIF dengan kampanye Amazon Personalize. Anda menggunakan kampanye untuk mendapatkan rekomendasi bagi pengguna Anda. Lihat Menerapkan versi solusi Amazon Personalize dengan kampanye.
- Untuk rekomendasi batch, Anda menentukan versi solusi AKTIF saat membuat pekerjaan inferensi batch atau pekerjaan segmen batch. Lihat Mendapatkan rekomendasi item batch atau Mendapatkan segmen pengguna batch.

Membuat versi solusi (AWS SDKs)

Ketika solusi Anda AKTIF, gunakan kode berikut untuk membuat versi solusi. Tentukan Nama Sumber Daya Amazon (ARN) dari. Mengonfigurasi solusi khusus di Amazon Personalize Gunakan DescribeSolutionVersion operasi untuk mengambil status versi solusi.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')
# Store the solution ARN
solution_arn = 'solution arn'

# Use the solution ARN to get the solution status.
solution_description = personalize.describe_solution(solutionArn = 'solution_arn')
['solution']
print('Solution status: ' + solution_description['status'])

# Use the solution ARN to create a solution version.
print ('Creating solution version')
response = personalize.create_solution_version(solutionArn = solution_arn)
solution_version_arn = response['solutionVersionArn']
print('Solution version ARN: ' + solution_version_arn)

# Use the solution version ARN to get the solution version status.
solution_version_description = personalize.describe_solution_version(
    solutionVersionArn = solution_version_arn)['solutionVersion']
print('Solution version status: ' + solution_version_description['status'])
```

SDK for Java 2.x

```
public static String createPersonalizeSolutionVersion(PersonalizeClient
 personalizeClient, String solutionArn) {
```

```
        long maxTime = 0;
        long waitInMilliseconds = 30 * 1000; // 30 seconds
        String solutionStatus = "";
        String solutionVersionStatus = "";
        String solutionVersionArn = "";

        try {
            DescribeSolutionRequest describeSolutionRequest =
DescribeSolutionRequest.builder()
                .solutionArn(solutionArn)
                .build();

            maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            // Wait until solution is active.
            while (Instant.now().getEpochSecond() < maxTime) {

                solutionStatus =
personalizeClient.describeSolution(describeSolutionRequest).solution().status();
                System.out.println("Solution status: " + solutionStatus);

                if (solutionStatus.equals("ACTIVE") || solutionStatus.equals("CREATE
FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }

            // Once the solution is active, start creating a solution version.

            if (solutionStatus.equals("ACTIVE")) {

                CreateSolutionVersionRequest createSolutionVersionRequest =
CreateSolutionVersionRequest.builder()
                    .solutionArn(solutionArn)
                    .build();

                CreateSolutionVersionResponse createSolutionVersionResponse =
personalizeClient.createSolutionVersion(createSolutionVersionRequest);
```

```
                solutionVersionArn =
createSolutionVersionResponse.solutionVersionArn();

                System.out.println("Solution version ARN: " + solutionVersionArn);

                DescribeSolutionVersionRequest describeSolutionVersionRequest =
DescribeSolutionVersionRequest.builder()
                    .solutionVersionArn(solutionVersionArn)
                    .build();

                maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

                while (Instant.now().getEpochSecond() < maxTime) {

                    // Use the solution version ARN to get the solution version
status.
                    solutionVersionStatus =
personalizeClient.describeSolutionVersion(describeSolutionVersionRequest).solutionVersion()
                    System.out.println("Solution version status: " +
solutionVersionStatus);

                    if (solutionVersionStatus.equals("ACTIVE") ||
solutionVersionStatus.equals("CREATE FAILED")) {
                        break;
                    }
                    try {
                        Thread.sleep(waitInMilliseconds);
                    } catch (InterruptedException e) {
                        System.out.println(e.getMessage());
                    }
                }
                return solutionVersionArn;
            }
        } catch(PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateSolutionVersionCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the solution version parameters.
export const solutionVersionParam = {
  solutionArn: "SOLUTION_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSolutionVersionCommand(solutionVersionParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Untuk memeriksa status versi solusi saat ini, panggil DescribeSolutionVersion operasi dan lewati ARN dari versi solusi yang dikembalikan dari operasi. `CreateSolutionVersion` Pelatihan selesai saat `status` selesai `ACTIVE` dan Anda dapat mengevaluasinya menggunakan metrik yang disediakan oleh Amazon Personalize. Untuk informasi selengkapnya, lihat Mengevaluasi versi solusi Amazon Personalize dengan metrik. Jika pelatihan tidak selesai karena kesalahan, Anda tidak dikenakan biaya untuk pelatihan.

Jika versi solusi Anda memiliki status CREATE_PENDING atau CREATE_IN_PROGRESS, Anda dapat menggunakan StopSolutionVersionCreation operasi untuk menghentikan proses pembuatan versi solusi. Lihat Menghentikan pembuatan versi solusi.

Ketika versi solusi AKTIF, Anda siap menggunakannya untuk mendapatkan rekomendasi. Bagaimana Anda menggunakan versi solusi aktif tergantung pada bagaimana Anda mendapatkan rekomendasi:

- Untuk rekomendasi real-time, Anda menerapkan versi solusi AKTIF dengan kampanye Amazon Personalize. Anda menggunakan kampanye untuk mendapatkan rekomendasi bagi pengguna Anda. Lihat Menerapkan versi solusi Amazon Personalize dengan kampanye.
- Untuk rekomendasi batch, Anda menentukan versi solusi AKTIF saat membuat pekerjaan inferensi batch atau pekerjaan segmen batch. Lihat Mendapatkan rekomendasi item batch atau Mendapatkan segmen pengguna batch.

Menghentikan pembuatan versi solusi

Jika versi solusi Anda memiliki status CREATE_PENDING atau CREATE_IN_PROGRESS, Anda dapat menggunakan konsol Amazon Personalize atau StopSolutionVersionCreation operasi untuk berhenti membuat versi solusi (berhenti melatih model). Anda tidak dapat melanjutkan pembuatan versi solusi setelah berhenti. Anda ditagih untuk sumber daya yang digunakan hingga saat pembuatan versi solusi dihentikan.

Menghentikan pembuatan versi solusi mengakhiri pelatihan model, tetapi tidak menghapus versi solusi. Anda masih dapat melihat detail versi solusi di konsol Amazon Personalize dan dengan operasi. DescribeSolutionVersion

Anda dapat menghentikan proses pembuatan versi solusi dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs

Topik

- Menghentikan pembuatan versi solusi (konsol)
- Menghentikan pembuatan versi solusi (AWS CLI)
- Menghentikan pembuatan versi solusi (AWS SDKs)

Menghentikan pembuatan versi solusi (konsol)

Jika versi solusi Anda memiliki status CREATE_PENDING atau CREATE_IN_PROGRESS, Anda dapat berhenti membuat versi solusi (berhenti melatih model).

Untuk berhenti membuat versi solusi (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.

2. Pada halaman grup Dataset, pilih grup kumpulan data dengan versi solusi yang ingin Anda hentikan.
3. Di panel navigasi, pilih Solusi dan resep.
4. Pada halaman Solusi dan resep, pilih solusi dengan versi solusi yang ingin Anda hentikan.
5. Dalam versi Solusi, pilih versi solusi yang ingin Anda hentikan.
6. Pada halaman detail versi solusi, pilih Hentikan pembuatan. Bergantung pada status asli versi solusi, status versi solusi berubah sebagai berikut:

   - CREATE_PENDING berubah menjadi CREATE_STOPPED.
   - CREATE_IN_PROGRESS berubah menjadi CREATE_STOPING dan kemudian CREATE_STOPPED.

Menghentikan pembuatan versi solusi (AWS CLI)

Jika versi solusi Anda memiliki status CREATE_PENDING atau CREATE_IN_PROGRESS, Anda dapat berhenti membuat versi solusi (berhenti melatih model). Gunakan `stop-solution-version-creation` perintah berikut untuk berhenti membuat versi solusi dengan AWS CLI. Ganti `solution version arn` dengan Amazon Resource Name (ARN) dari versi solusi yang ingin Anda hentikan. Anda ditagih untuk sumber daya yang digunakan hingga pembuatan versi solusi dihentikan.

```
aws personalize stop-solution-version-creation \
    --solution-version-arn solution version arn
```

Periksa status pelatihan versi solusi dengan `describe-solution-version` perintah.

```
aws personalize describe-solution-version \
    --solution-version-arn solution version arn
```

Bergantung pada status asli versi solusi, status versi solusi berubah sebagai berikut:

- CREATE_PENDING berubah menjadi CREATE_STOPPED.
- CREATE_IN_PROGRESS berubah menjadi CREATE_STOPING dan kemudian CREATE_STOPPED

Menghentikan pembuatan versi solusi (AWS SDKs)

Jika versi solusi Anda memiliki status CREATE_PENDING atau CREATE_IN_PROGRESS, Anda dapat berhenti membuat versi solusi (berhenti melatih model). Kode berikut menunjukkan cara berhenti membuat versi solusi dengan AWS SDK untuk Python (Boto3) atau AWS SDK for Java 2.x. Anda ditagih untuk sumber daya yang digunakan hingga saat pembuatan versi solusi dihentikan.

SDK for Python (Boto3)

Gunakan `stop_solution_version_creation` metode berikut untuk menghentikan pembuatan versi solusi. Ganti `solution_version_arn` dengan Amazon Resource Name (ARN) dari versi solusi yang ingin Anda hentikan. Metode ini menggunakan DescribeSolutionVersion operasi untuk mengambil status versi solusi.

```
import boto3

personalize = boto3.client('personalize')

response = personalize.stop_solution_version_creation(
    solutionVersionArn = solution_version_arn
)

# Use the solution version ARN to get the solution version status.
solution_version_description = personalize.describe_solution_version(
    solutionVersionArn = solution_version_arn)['solutionVersion']
print('Solution version status: ' + solution_version_description['status'])
```

SDK for Java 2.x

Gunakan `stopSolutionVersionCreation` metode berikut untuk berhenti membuat versi solusi. Teruskan sebagai parameter klien layanan Amazon Personalisasi dan Nama Sumber Daya Amazon (ARN) dari versi solusi yang ingin Anda hentikan pembuatannya. Kode berikut menggunakan DescribeSolutionVersion operasi untuk mengambil status versi solusi.

```
public static void stopSolutionVersionCreation(PersonalizeClient personalizeClient,
 String solutionVersionArn) {
    String solutionVersionStatus = "";

    StopSolutionVersionCreationRequest stopSolutionVersionCreationRequest =
 StopSolutionVersionCreationRequest.builder()
```

```
        .solutionVersionArn(solutionVersionArn)
        .build();


 personalizeClient.stopSolutionVersionCreation(stopSolutionVersionCreationRequest);

    // Use the solution version ARN to get the solution version status.
    DescribeSolutionVersionRequest describeSolutionVersionRequest =
 DescribeSolutionVersionRequest.builder()
        .solutionVersionArn(solutionVersionArn)
        .build();

    solutionVersionStatus =
 personalizeClient.describeSolutionVersion(describeSolutionVersionRequest)
        .solutionVersion()
        .status();
    System.out.println("Solution version status: " + solutionVersionStatus);
}
```

Bergantung pada status asli versi solusi, status versi solusi berubah sebagai berikut:

- CREATE_PENDING berubah menjadi CREATE_STOPPED.
- CREATE_IN_PROGRESS berubah menjadi CREATE_STOPING dan kemudian CREATE_STOPPED.

Mengevaluasi versi solusi Amazon Personalize dengan metrik

Anda dapat mengevaluasi kinerja versi solusi Anda melalui metrik offline dan online. Metrik online adalah hasil empiris yang Anda amati dalam interaksi pengguna Anda dengan rekomendasi waktu nyata. Misalnya, Anda dapat merekam rasio klik-tayang pengguna saat mereka menelusuri katalog Anda. Anda bertanggung jawab untuk membuat dan merekam metrik online apa pun.

Metrik offline adalah metrik yang dihasilkan Amazon Personalize saat Anda melatih versi solusi. Dengan metrik offline, Anda dapat mengevaluasi kinerja model. Anda dapat melihat efek memodifikasi hiperparameter solusi, dan Anda dapat membandingkan hasil dari model yang dilatih dengan resep berbeda pada data yang sama dalam grup kumpulan data yang sama.

Hindari membandingkan metrik versi solusi berbeda yang dilatih dengan data yang berbeda. Perbedaan metrik mungkin berasal dari perbedaan data daripada kinerja model. Misalnya,

Anda mungkin memiliki grup kumpulan data dengan data `purchase` peristiwa jarang untuk setiap pengguna, dan grup lainnya dengan data `view` peristiwa yang kuat. Berdasarkan metrik seperti`precision at K`, versi solusi yang dilatih pada data peristiwa tampilan mungkin salah tampak berkinerja lebih baik karena jumlah interaksi yang lebih tinggi.

Untuk mendapatkan metrik performa, Amazon Personalize membagi data interaksi input menjadi set pelatihan, set pengujian, dan untuk PERSONALIZED_ACTIONS, set validasi. Pembagian tergantung pada jenis resep yang Anda pilih:

- Untuk resep USER_SEGMENTATION, set pelatihan terdiri dari 80% data interaksi setiap pengguna dan set pengujian terdiri dari 20% dari data interaksi setiap pengguna.
- Untuk semua jenis resep lainnya, set pelatihan terdiri dari 90% pengguna Anda dan data interaksinya. Set pengujian terdiri dari 10% sisa pengguna dan data interaksi mereka.

Amazon Personalize kemudian membuat versi solusi menggunakan set pelatihan. Setelah pelatihan selesai, Amazon Personalize memberikan versi solusi baru 90% tertua dari setiap data pengguna dari set pengujian sebagai input. Amazon Personalize kemudian menghitung metrik dengan membandingkan rekomendasi yang dihasilkan versi solusi dengan interaksi aktual dalam 10% terbaru dari setiap data pengguna dari set pengujian.

Untuk menghasilkan garis dasar untuk tujuan perbandingan, kami sarankan menggunakan Hitungan Popularitas resep, yang merekomendasikan item K paling populer teratas.

Topik

- Mengambil metrik versi solusi
- Definisi metrik
- Contoh
- Sumber daya tambahan

Mengambil metrik versi solusi

Setelah membuat versi solusi, Anda dapat menggunakan metrik untuk mengevaluasi kinerjanya. Anda dapat mengambil metrik untuk versi solusi dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), dan. AWS SDKs

Topik

- Mengambil metrik versi solusi (konsol)
- Mengambil metrik versi solusi ()AWS CLI
- Mengambil metrik versi solusi ()AWS SDKs

Mengambil metrik versi solusi (konsol)

Untuk melihat metrik pemberi rekomendasi di konsol, Anda menavigasi ke halaman detail untuk versi solusi Anda.

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup Dataset Kustom Anda.
3. Dari panel navigasi, pilih Sumber daya khusus, lalu pilih Solusi dan resep.
4. Pilih solusi Anda.
5. Dalam versi Solusi, pilih versi solusi Anda untuk melihat halaman detailnya. Metrik tercantum pada tab Metrik versi Solusi di panel bawah. Untuk definisi metrik, lihatDefinisi metrik.

   Setelah mengevaluasi versi solusi, Anda dapat membuat kampanye dengan menerapkan versi solusi dengan metrik terbaik untuk kasus penggunaan Anda. Untuk informasi selengkapnya tentang penerapan solusi, lihatMenerapkan versi solusi Amazon Personalize dengan kampanye.

Mengambil metrik versi solusi ()AWS CLI

Anda mengambil metrik untuk versi solusi tertentu dengan memanggil operasi. GetSolutionMetrics Kode berikut menunjukkan cara mengambil metrik dengan. AWS CLI

```
personalize get-solution-metrics --solution-version-arn solution version ARN
```

Berikut ini adalah contoh output dari versi solusi yang dibuat menggunakan Personalisasi Pengguna resep dengan tujuan optimasi tambahan.

```
{
    "solutionVersionArn": "arn:aws:personalize:us-west-2:acct-id:solution/SolutionName/
<version-id>",
    "metrics": {
        "coverage": 0.27,
        "mean_reciprocal_rank_at_25": 0.0379,
```

```
        "normalized_discounted_cumulative_gain_at_5": 0.0405,
        "normalized_discounted_cumulative_gain_at_10": 0.0513,
        "normalized_discounted_cumulative_gain_at_25": 0.0828,
        "precision_at_5": 0.0136,
        "precision_at_10": 0.0102,
        "precision_at_25": 0.0091,
        "average_rewards_at_k": 0.653
    }
}
```

Untuk penjelasan dari setiap metrik, lihatDefinisi metrik. Setelah mengevaluasi versi solusi, Anda dapat membuat kampanye dengan menerapkan versi solusi dengan metrik terbaik untuk kasus penggunaan Anda. Untuk informasi selengkapnya tentang penerapan solusi, lihatMenerapkan versi solusi Amazon Personalize dengan kampanye.

Mengambil metrik versi solusi ()AWS SDKs

Anda mengambil metrik untuk versi solusi tertentu dengan memanggil operasi. GetSolutionMetrics Gunakan kode berikut untuk mengambil metrik.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.get_solution_metrics(
    solutionVersionArn = 'solution version arn')

print(response['metrics'])
```

SDK for Java 2.x

```
public static void getSolutionVersionMetrics(PersonalizeClient personalizeClient,
 String solutionVersionArn) {

    try {
        GetSolutionMetricsRequest request = GetSolutionMetricsRequest.builder()
                .solutionVersionArn(solutionVersionArn)
                .build();
        Map<String, Double> metrics =
 personalizeClient.getSolutionMetrics(request).metrics();
```

```
        metrics.forEach((key, value) -> System.out.println(key + " " + value));
    } catch (PersonalizeException e ) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
}
```

Berikut ini adalah contoh output dari versi solusi yang dibuat menggunakan Personalisasi Pengguna resep dengan tujuan optimasi tambahan.

```
{
    "solutionVersionArn": "arn:aws:personalize:us-west-2:acct-id:solution/
MovieSolution/<version-id>",
    "metrics": {
        "coverage": 0.27,
        "mean_reciprocal_rank_at_25": 0.0379,
        "normalized_discounted_cumulative_gain_at_5": 0.0405,
        "normalized_discounted_cumulative_gain_at_10": 0.0513,
        "normalized_discounted_cumulative_gain_at_25": 0.0828,
        "precision_at_5": 0.0136,
        "precision_at_10": 0.0102,
        "precision_at_25": 0.0091,
        "average_rewards_at_k": 0.653
    }
}
```

Untuk penjelasan dari setiap metrik, lihatDefinisi metrik. Setelah mengevaluasi versi solusi, Anda dapat membuat kampanye dengan menerapkan versi solusi dengan metrik terbaik untuk kasus penggunaan Anda. Untuk informasi selengkapnya tentang penerapan solusi, lihatMenerapkan versi solusi Amazon Personalize dengan kampanye.

Definisi metrik

Metrik yang dihasilkan Amazon Personalize untuk versi solusi dijelaskan di bawah ini menggunakan istilah berikut:

- Rekomendasi yang relevan adalah rekomendasi untuk item yang benar-benar berinteraksi dengan pengguna. Item ini berasal dari 10% terbaru dari data interaksi setiap pengguna dari set pengujian.
- Peringkat mengacu pada posisi item yang direkomendasikan dalam daftar rekomendasi. Posisi 1 (bagian atas daftar) dianggap paling relevan bagi pengguna.

Untuk setiap metrik, angka yang lebih tinggi (mendekati 1) lebih baik. Untuk menyelam lebih dalam, lihat sumber daya yang tercantum diSumber daya tambahan.

cakupan

Nilai cakupan memberi tahu Anda proporsi item unik (untuk rekomendasi item), tindakan (untuk rekomendasi tindakan), atau pengguna (untuk rekomendasi segmen pengguna), yang mungkin direkomendasikan Amazon Personalize dari jumlah total catatan unik dalam kumpulan data Anda.

Skor cakupan yang lebih tinggi berarti Amazon Personalize merekomendasikan lebih banyak katalog Anda, daripada catatan yang sama berulang kali. Resep yang menampilkan eksplorasi item, seperti Personalisasi Pengguna, memiliki cakupan yang lebih tinggi daripada yang tidak, seperti Item Serupa.

rata-rata peringkat timbal balik pada 25

Metrik ini memberi tahu Anda tentang kemampuan model untuk menghasilkan rekomendasi item yang relevan di posisi peringkat teratas.

Anda dapat memilih model dengan peringkat timbal balik rata-rata tinggi di 25 jika Anda menghasilkan hasil pencarian item untuk pengguna, dan jangan berharap pengguna memilih item yang lebih rendah dalam daftar. Misalnya, pengguna sering memilih resep memasak pertama di hasil pencarian. Amazon Personalize tidak menghasilkan metrik ini untuk resep PERSONALIZED_ACTIONS atau USER_SEGMENTATION.

Amazon Personalize menghitung metrik ini menggunakan skor peringkat timbal balik rata-rata untuk permintaan rekomendasi. Setiap skor peringkat timbal balik dihitung sebagai berikut:`1 / the rank of the highest item interacted with by the user`, di mana total peringkat yang mungkin adalah 25. Item berperingkat rendah lainnya yang berinteraksi dengan pengguna diabaikan. Jika pengguna memilih item pertama, skornya adalah 1. Jika mereka tidak memilih item apa pun, skornya adalah 0.

Misalnya, Anda mungkin menampilkan tiga pengguna yang berbeda 25 rekomendasi masing-masing:

- Jika Pengguna 1 mengklik item di peringkat 4 dan item di peringkat 10, skor peringkat timbal balik mereka adalah 1/4.
- Jika Pengguna 2 mengklik item di peringkat 2, item di peringkat 4, dan item di peringkat 12, skor peringkat timbal balik mereka adalah 1/2.
- Jika Pengguna 3 mengklik satu item di peringkat 6, skor peringkat timbal balik mereka adalah 1/6.

Peringkat timbal balik rata-rata atas semua permintaan rekomendasi (dalam hal ini 3) dihitung sebagai. `(1/4 + 1/2 + 1/6) / 3 = .3056`

keuntungan kumulatif diskon yang dinormalisasi (NDCG) di K (5/10/25)

Metrik ini memberi tahu Anda tentang seberapa baik model Anda memberi peringkat item atau rekomendasi tindakan, di mana K adalah ukuran sampel 5, 10, atau 25 rekomendasi. Metrik ini berguna jika Anda paling tertarik dengan peringkat rekomendasi di luar hanya item atau tindakan peringkat tertinggi (untuk ini, lihat`mean reciprocal rank at 25`). Misalnya, skor untuk `NDCG at 10` akan berguna jika Anda memiliki aplikasi yang menampilkan hingga 10 film dalam korsel sekaligus.

Amazon Personalize menghitung NDCG dengan menetapkan bobot rekomendasi berdasarkan posisi peringkat mereka untuk setiap pengguna dalam set pengujian. Setiap rekomendasi didiskon (diberi bobot yang lebih rendah) oleh faktor yang bergantung pada posisinya. Metrik akhir adalah rata-rata `NDCG at K` untuk semua pengguna di set pengujian. `NDCG at K`Mengasumsikan bahwa rekomendasi yang lebih rendah dalam daftar kurang relevan daripada rekomendasi yang lebih tinggi dalam daftar.

Amazon Personalize menggunakan faktor pembobotan`1/log(1 + position)`, di mana bagian atas daftar adalah posisi. `1`

presisi di K

Metrik ini memberi tahu Anda seberapa relevan rekomendasi model Anda berdasarkan ukuran sampel rekomendasi K (5, 10, atau 25).

Amazon Personalize menghitung metrik ini berdasarkan jumlah rekomendasi yang relevan dari rekomendasi K teratas untuk setiap pengguna dalam set pengujian, dibagi dengan K, di mana K adalah 5, 10, atau 25. Metrik akhir adalah rata-rata di semua pengguna dalam set pengujian.

Misalnya, jika Anda merekomendasikan 10 item kepada pengguna, dan pengguna berinteraksi dengan 3 item, presisi di K adalah 3 item yang diprediksi dengan benar dibagi dengan total 10 item yang direkomendasikan:`3 / 10 = .30`.

Metrik ini menghargai rekomendasi yang tepat dari item yang relevan. Semakin dekat skornya ke satu, semakin tepat modelnya.

ketepatan

Jika Anda melatih versi solusi dengan Next-Best-Action resep, Amazon Personalize menghasilkan `precision` metrik, bukan. `precision at K` Metrik ini memberi tahu Anda seberapa bagus model Anda dalam memprediksi tindakan yang sebenarnya akan diambil pengguna.

Untuk menghitung`precision`, untuk setiap tindakan dalam kumpulan data Anda, Amazon Personalize membagi jumlah pengguna yang diprediksi dengan benar untuk mengambil tindakan dengan jumlah kali tindakan direkomendasikan. Amazon Personalize kemudian menghitung rata-rata untuk semua tindakan dalam kumpulan data Anda.

Misalnya, jika tindakan direkomendasikan untuk 100 pengguna, dan 60 pengguna mengambil tindakan dan 40 pengguna yang tidak, tindakan tersebut adalah:`60 / 100 = .60`. `precision` Amazon Personalize kemudian menerapkan perhitungan ini untuk semua tindakan dan mengembalikan rata-rata.

Metrik ini menghargai rekomendasi yang tepat dari tindakan yang relevan. Semakin dekat skornya ke satu, semakin tepat modelnya.

rata-rata_rewards_at_k

Saat Anda membuat versi solusi (melatih model) untuk solusi dengan tujuan pengoptimalan, Amazon Personalize menghasilkan metrik. `average_rewards_at_k` Skor untuk `average_rewards_at_k` memberi tahu Anda seberapa baik kinerja versi solusi dalam mencapai tujuan Anda. Untuk menghitung metrik ini, Amazon Personalize menghitung hadiah untuk setiap pengguna sebagai berikut:

```
rewards_per_user = total rewards from the user's interactions with their top 25 reward generating recommendations / total rewards from the user's interactions with recommendations
```

Final `average_rewards_at_k` adalah rata-rata semua `rewards_per_user` dinormalisasi menjadi nilai desimal kurang dari atau sama dengan 1 dan lebih besar dari 0. Semakin dekat nilainya ke 1, semakin banyak keuntungan rata-rata per pengguna yang dapat Anda harapkan dari rekomendasi.

Misalnya, jika tujuan Anda adalah memaksimalkan pendapatan dari klik, Amazon Personalize menghitung setiap skor pengguna dengan membagi total pendapatan yang dihasilkan oleh item yang diklik pengguna dari 25 rekomendasi termahal teratas mereka dengan pendapatan dari semua item yang direkomendasikan yang diklik pengguna. Amazon Personalize kemudian mengembalikan rata-rata yang dinormalisasi dari semua skor pengguna. Semakin

`average_rewards_at_k` dekat ke 1, semakin banyak pendapatan rata-rata yang dapat Anda peroleh per pengguna dari rekomendasi.

Untuk informasi selengkapnya, lihat Mengoptimalkan solusi untuk tujuan tambahan.

normalized_discounted_cumulative_gain_with_event_weights_at_k

Saat Anda membuat versi solusi (melatih model) untuk solusi dengan konfigurasi peristiwa, Amazon Personalize akan menghasilkan metrik. `normalized_discounted_cumulative_gain_with_event_weights_at_k` Skor untuk `normalized_discounted_cumulative_gain_with_event_weights_at_k` memberi tahu Anda seberapa baik kinerja versi solusi mengingat bobot acara yang Anda tetapkan untuk setiap jenis acara.

Ini mirip dengan keuntungan kumulatif diskon yang dinormalisasi (NDCG) di K tetapi hadiah untuk setiap prediksi yang benar akan tertimbang. Sebaliknya, dalam NDCG asli di K, setiap prediksi yang benar semuanya akan membawa bobot 1. Misalnya, dengan “pembelian” berat 0,3 dan “klik” berat 0,1, memprediksi dengan benar item “pembelian” akan mendapatkan hadiah 1,5 sementara memprediksi item “klik” akan mendapatkan hadiah 0,5.

Untuk informasi selengkapnya, lihat Mengoptimalkan solusi dengan konfigurasi peristiwa.

akurasi prediksi tren

Jika Anda melatih versi solusi dengan Trend-Sekarang resep, tingkat peningkatan popularitas item yang direkomendasikan oleh model. Semakin tinggi akurasi prediksi tren (semakin dekat ke 1), semakin baik model dalam mengidentifikasi item yang sedang tren dengan benar.

Untuk menghitung akselerasi popularitas, Amazon Personalize membagi tingkat peningkatan popularitas di semua item yang direkomendasikan dengan peningkatan popularitas total dari 25 item tren teratas. Item ini berasal dari interaksi aktual dalam set pengujian.

Bergantung pada distribusi data Anda dan apa yang Anda pilih untuk frekuensi penemuan Trend, nilai akurasi prediksi tren bisa 0,0.

memukul (memukul di K)

Jika Anda melatih versi solusi dengan resep USER_SEGMENTATION, jumlah rata-rata pengguna dalam hasil K relevan teratas yang diprediksi akan cocok dengan pengguna sebenarnya. Pengguna sebenarnya adalah pengguna yang benar-benar berinteraksi dengan item dalam set pengujian. K adalah 1% teratas dari pengguna yang paling relevan. Semakin tinggi nilainya semakin akurat prediksi.

ingat (ingat di K)

Jika Anda melatih versi solusi dengan resep USER_SEGMENTATION, persentase rata-rata pengguna yang diprediksi dalam hasil K relevan teratas yang diprediksi yang cocok dengan pengguna sebenarnya. Pengguna sebenarnya adalah pengguna yang benar-benar berinteraksi dengan item dalam set pengujian. K adalah 1% teratas dari pengguna yang paling relevan. Semakin tinggi nilainya, semakin akurat prediksi.

penarikan

Jika Anda melatih versi solusi dengan Next-Best-Action resep, metrik ini memberi tahu Anda seberapa baik versi solusi Anda dalam menemukan tindakan yang akan berinteraksi dengan pengguna.

Untuk menghitung`recall`, untuk setiap tindakan dalam kumpulan data Anda, Amazon Personalize membagi jumlah pengguna yang diprediksi dengan benar untuk mengambil tindakan dengan jumlah total pengguna yang benar-benar mengambil tindakan dalam set pengujian. Amazon Personalize kemudian menghitung rata-rata untuk semua tindakan dalam kumpulan data Anda.

Misalnya, jika 100 pengguna mengambil tindakan dalam set pengujian, dan Amazon Personalize memperkirakan 50 pengguna ini akan mengambil tindakan, tindakan `recall` untuk tindakan tersebut adalah: `50 / 100 = .50` Amazon Personalize kemudian menerapkan perhitungan ini untuk semua tindakan dan mengembalikan rata-rata.

Area di bawah kurva (AUC)

Jika Anda melatih versi solusi dengan resep PERSONALIZED_ACTIONS, area di bawah kurva Karakteristik Operasi Penerima untuk versi solusi Anda. Metrik ini memberi tahu Anda seberapa baik kinerja versi solusi dalam mengidentifikasi tindakan yang akan dilakukan pengguna dengan benar.

Kurva Karakteristik Operasi Penerima memplot kinerja versi solusi. Ini memplot tingkat positif sejati (tindakan yang diprediksi dengan benar sebagai relevan) dan positif palsu (tindakan yang salah diprediksi sebagai relevan) pada nilai ambang batas yang berbeda. Area di bawah kurva (AUC) adalah skor yang merangkum kinerja versi solusi berdasarkan kurvanya.

AUC dari versi solusi dapat antara 0 dan 1. Semakin dekat ke 1, semakin baik model dalam memprediksi tindakan yang relevan untuk pengguna Anda.

Contoh

Berikut ini adalah contoh sederhana untuk versi solusi yang menghasilkan daftar rekomendasi untuk pengguna tertentu. Rekomendasi kedua dan kelima cocok dengan catatan dalam data pengujian untuk pengguna ini. Ini adalah rekomendasi yang relevan. Jika K diatur pada5, metrik berikut dihasilkan untuk pengguna.

reciprocal_rank

Perhitungan: 1/2

Hasil: 0.5000

normalized_discounted_cumulative_gain_at_5

Perhitungan: (1/log (1 + 2) + 1/log (1 + 5))/(1/log (1 + 1) + 1/log (1 + 2))

Hasil: 0.6241

presisi_at_5

Perhitungan: 2/5

Hasil: 0.4000

Sumber daya tambahan

Untuk informasi tentang mengevaluasi versi solusi dengan A/B pengujian, lihat Menggunakan A/B pengujian untuk mengukur kemanjuran rekomendasi yang dihasilkan oleh Amazon Personalize. Untuk menyelami lebih dalam berbagai jenis metrik untuk sistem pemberi rekomendasi, lihat sumber daya eksternal berikut:

- MRR vs MAP vs NDCG: Metrik Evaluasi Sadar Peringkat Dan Kapan Menggunakannya
- Keuntungan Kumulatif Diskon: metrik peringkat yang harus Anda ketahui
- Recall dan Presisi di k untuk Sistem Recommender
- Metrik Evaluasi Peringkat untuk Sistem Rekomendasi
- Karakteristik operasi penerima

Menerapkan versi solusi Amazon Personalize dengan kampanye

Untuk rekomendasi real-time dengan sumber daya khusus, setelah selesaiMembuat versi solusi secara manual, Anda siap untuk menerapkan versi solusi Anda dengan kampanye.

Kampanye menyebarkan versi solusi (model terlatih) dengan kapasitas transaksi yang disediakan untuk menghasilkan rekomendasi waktu nyata. Setelah membuat kampanye, Anda menggunakan operasi GetRecommendations atau GetPersonalizedRanking API untuk mendapatkan rekomendasi. Jika Anda mendapatkan rekomendasi item batch atau segmen pengguna, Anda tidak perlu membuat kampanye. Untuk informasi selengkapnya, lihat Mendapatkan rekomendasi item batch atau Mendapatkan segmen pengguna batch.

Saat membuat kampanye, Anda dapat mengonfigurasi hal berikut:

- Anda dapat mengonfigurasi kampanye untuk memperbarui secara otomatis untuk menggunakan versi solusi terbaru solusi Anda. Untuk mengetahui informasi selengkapnya, lihat Pembaruan kampanye otomatis.
- Anda dapat mengaktifkan metadata item dalam rekomendasi. Untuk informasi selengkapnya, lihat Metadata item dalam rekomendasi.
- Anda dapat menentukan transaksi minimum yang disediakan per detik untuk kampanye. Ini adalah throughput transaksi dasar untuk kampanye yang disediakan oleh Amazon Personalize. Ini menetapkan biaya penagihan minimum untuk kampanye saat sedang aktif. Untuk informasi selengkapnya, lihat Minimum transaksi yang disediakan per detik dan auto-scaling.

Anda dapat membuat kampanye dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs Jika ingin mengubah setelan kampanye yang ada, seperti mengaktifkan metadata dalam rekomendasi, Anda harus memperbarui kampanye. Untuk informasi selengkapnya, lihat Memperbarui konfigurasi kampanye Amazon Personalize.

Anda dikenakan biaya kampanye saat kampanye aktif. Untuk menghindari biaya yang tidak perlu, pastikan untuk menghapus kampanye ketika Anda selesai. Untuk informasi tentang biaya kampanye, lihat harga Amazon Personalisasi.

Topik

- Pembaruan kampanye otomatis
- Minimum transaksi yang disediakan per detik dan auto-scaling
- Metadata item dalam rekomendasi

- Membuat kampanye (konsol)
- Membuat kampanye (AWS CLI)
- Membuat kampanye (AWS SDKs)

Pembaruan kampanye otomatis

Saat membuat kampanye, Anda dapat mengaktifkan pembaruan kampanye otomatis. Dengan pembaruan otomatis, kampanye secara otomatis memperbarui untuk menerapkan versi solusi terbaru yang terlatih secara otomatis atau manual dari solusi Anda. Ini memudahkan Anda untuk menjaga kampanye Anda tetap terkini.

Misalnya, jika solusi Anda menggunakan pelatihan otomatis untuk membuat versi solusi baru setiap tujuh hari, kampanye Anda akan diperbarui secara otomatis untuk menggunakan versi solusi terbaru untuk setiap pelatihan mingguan. Jika Anda tidak menggunakan pembaruan kampanye otomatis, Anda harus memperbarui kampanye secara manual untuk menerapkan model terlatih terbaru.

- Untuk mengaktifkan pembaruan kampanye otomatis saat Anda membuat kampanye dengan konsol Amazon Personalisasi, pilih Perbarui secara otomatis untuk menggunakan versi solusi terbaru solusi Anda di detail Kampanye. Anda dapat menemukan stempel waktu untuk pembaruan terbaru di halaman detail kampanye.

  Untuk informasi selengkapnya, lihat Membuat kampanye (konsol).
- Untuk mengaktifkan pembaruan kampanye otomatis saat Anda menggunakan operasi CreateCampaign API, untuk `SolutionVersionArn` parameter, tentukan Nama Sumber Daya Amazon (ARN) solusi Anda dalam `SolutionArn/$LATEST` format. Di`campaignConfig`, atur `enableMetadataWithRecommendations` ke`true`.

  Untuk mendapatkan stempel waktu pembaruan kampanye terbaru, Anda dapat menggunakan operasi DescribeCampaign API dan memeriksa `latestCampaignUpdate` detail dalam respons.

  Untuk contoh kode yang menunjukkan cara mengaktifkan pembaruan otomatis, lihat Membuat kampanye (AWS CLI) atauMembuat kampanye (AWS SDKs).

Minimum transaksi yang disediakan per detik dan auto-scaling

> **Important**
>
> Tinggi `minProvisionedTPS` akan meningkatkan biaya Anda. Sebaiknya mulai dengan 1 untuk `minProvisionedTPS` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya.

Saat membuat kampanye Amazon Personalize, Anda dapat menentukan transaksi minimum yang disediakan per detik (`minProvisionedTPS`) untuk kampanye. Ini adalah throughput transaksi dasar untuk kampanye yang disediakan oleh Amazon Personalize. Ini menetapkan biaya penagihan minimum untuk kampanye saat sedang aktif. Transaksi adalah satu `GetRecommendations` atau `GetPersonalizedRanking` permintaan. `minProvisionedTPS`Defaultnya adalah 1.

Jika TPS Anda meningkat lebih dari it`minProvisionedTPS`, Amazon Personalize secara otomatis menskalakan kapasitas yang disediakan ke atas dan ke bawah, tetapi tidak pernah di bawah. `minProvisionedTPS` Ada penundaan waktu singkat sementara kapasitas meningkat yang dapat menyebabkan hilangnya transaksi. Ketika lalu lintas Anda berkurang, kapasitas kembali ke`minProvisionedTPS`.

Anda dikenakan biaya untuk TPS minimum yang disediakan atau, jika permintaan Anda melebihi, TPS yang `minProvisionedTPS` sebenarnya. TPS sebenarnya adalah jumlah permintaan rekomendasi yang Anda buat. Sebaiknya mulai dengan yang rendah`minProvisionedTPS`, lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, lalu tingkatkan `minProvisionedTPS` seperlunya.

Untuk informasi selengkapnya tentang biaya kampanye, lihat harga Amazon Personalisasi.

Metadata item dalam rekomendasi

> **Important**
>
> Jika Anda menggunakan resep User-Personalization-v 2 atau Personalized-Ranking-v 2, Anda tidak dikenakan biaya tambahan untuk metadata. Untuk semua resep lain dan semua kasus penggunaan domain, Anda dikenakan biaya tambahan. Untuk informasi selengkapnya, lihat Harga Amazon Personalisasi.

Ketika Anda mendapatkan rekomendasi, Anda dapat meminta Amazon Personalize menyertakan metadata item dalam hasil rekomendasi. Dalam permintaan Anda, Anda dapat memilih kolom dari kumpulan data Item untuk disertakan. Amazon Personalize mengembalikan data ini untuk setiap item dalam respons rekomendasi.

Anda dapat menggunakan metadata untuk memperkaya rekomendasi di antarmuka pengguna, seperti menambahkan genre film ke komidi putar. Atau Anda dapat menggunakannya untuk menilai kualitas rekomendasi secara visual. Jika Anda menggunakan AI generatif di aplikasi, Anda dapat menyambungkan metadata ke prompt AI untuk menghasilkan konten yang lebih relevan. Untuk informasi selengkapnya tentang menggunakan Amazon Personalize dengan AI generatif, lihat. Amazon Personalisasi dan AI generatif

Mengaktifkan metadata

Untuk menambahkan metadata ke rekomendasi, Anda harus memiliki kumpulan data Item dengan kolom metadata. Anda tidak harus menggunakan metadata dalam pelatihan. Untuk informasi tentang membuat kumpulan data, lihatMembuat skema dan dataset. Untuk informasi yang memperbarui data, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Jika Anda menggunakan resep User-Personalization-v 2 atau Personalized-Ranking-v 2, kampanye baru secara otomatis memiliki opsi untuk menyertakan metadata item dengan hasil rekomendasi. Anda tidak mengaktifkan metadata secara manual untuk kampanye Anda. Untuk semua resep dan kasus penggunaan domain lainnya, Anda harus mengaktifkan opsi metadata:

- Untuk mengaktifkan metadata dengan konsol Amazon Personalize, saat Anda membuat kampanye, pilih Metadata pengembalian item di hasil rekomendasi di detail Kampanye. Untuk informasi selengkapnya, lihat Membuat kampanye (konsol).
- Untuk mengaktifkan metadata dengan AWS SDKs atau AWS CLI, gunakan operasi CreateCampaign API dan dalam `campaignConfig` set `enableMetadataWithRecommendations` ke. `true` Untuk informasi selengkapnya, lihat Membuat kampanye (AWS CLI) atau Membuat kampanye (AWS SDKs).

Membuat kampanye (konsol)

> **Important**
>
> Anda dikenakan biaya kampanye saat kampanye aktif. Untuk menghindari biaya yang tidak perlu, pastikan untuk menghapus kampanye ketika Anda selesai. Untuk informasi tentang biaya kampanye, lihat harga Amazon Personalisasi.

Setelah status versi solusi Aktif, Anda siap menerapkannya dengan kampanye Amazon Personalize.

Untuk membuat kampanye (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data dengan versi solusi yang ingin Anda terapkan.
3. Di panel navigasi, di bawah Sumber daya khusus, pilih Kampanye.
4. Pada halaman Kampanye, pilih Buat kampanye.
5. Pada halaman Buat kampanye baru, untuk detail Kampanye, berikan informasi berikut:
   - Nama kampanye — Masukkan nama kampanye. Teks yang Anda masukkan di sini muncul di dasbor Kampanye dan halaman detail.
   - Solusi — Pilih solusi yang baru saja Anda buat.
   - Perbarui secara otomatis untuk menggunakan versi solusi terbaru solusi Anda — Pilih opsi ini agar kampanye secara otomatis menggunakan versi solusi aktif terbaru. Jika Anda tidak memilih ini, Anda harus memperbarui kampanye secara manual setiap kali Anda ingin menerapkan versi solusi baru. Untuk informasi selengkapnya, lihat Pembaruan kampanye otomatis.
   - ID versi solusi - Jika Anda tidak menggunakan pembaruan kampanye otomatis untuk menggunakan versi solusi terbaru, pilih ID versi solusi yang ingin Anda gunakan.
   - Transaksi minimum yang disediakan per detik (disebut minProvisionedTPS di APIs) - Tetapkan transaksi minimum yang disediakan per detik yang didukung Amazon Personalize. Nilai tinggi akan meningkatkan biaya Anda. Kami menyarankan Anda memulai dengan 1 (default). Lacak penggunaan Anda dengan menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya. Untuk informasi selengkapnya, lihat Minimum transaksi yang disediakan per detik dan auto-scaling.

- Kembalikan metadata item dalam hasil rekomendasi — Pilih opsi ini jika Anda ingin opsi menyertakan metadata dengan hasil rekomendasi. Jika diaktifkan, Anda dapat menentukan kolom dari kumpulan data Item saat Anda mendapatkan rekomendasi. Untuk informasi selengkapnya, lihat Metadata item dalam rekomendasi.

6. Jika Anda menggunakan resep Personalisasi Pengguna, dalam konfigurasi Kampanye, Anda dapat secara opsional memasukkan nilai untuk bobot Eksplorasi dan batas usia item Eksplorasi. Untuk informasi selengkapnya, lihat Personalisasi Pengguna.
7. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya
8. Pilih Buat kampanye.
9. Pada halaman detail kampanye, ketika status kampanye Aktif, Anda dapat menggunakan kampanye untuk mendapatkan rekomendasi dan merekam tayangan. Untuk informasi selengkapnya, lihat Mendapatkan rekomendasi dari Amazon Personalize.

   Kampanye siap ketika statusnya AKTIF. Jika Anda melatih ulang versi solusi Anda, atau jika Anda ingin mengubah setelan kampanye, Anda harus memperbarui kampanye Anda. Untuk informasi selengkapnya, lihat Memperbarui konfigurasi kampanye Amazon Personalize.

Membuat kampanye (AWS CLI)

> **Important**
>
> Anda dikenakan biaya kampanye saat kampanye aktif. Untuk menghindari biaya yang tidak perlu, pastikan untuk menghapus kampanye ketika Anda selesai. Untuk informasi tentang biaya kampanye, lihat harga Amazon Personalisasi.

Setelah versi solusi Anda Aktif, Anda siap untuk menerapkannya dengan kampanye Amazon Personalize. Untuk membuat kampanye dengan AWS CLI, Anda menggunakan `create-campaign` perintah.

Contoh kode berikut menunjukkan cara membuat kampanye. Ini menyebarkan versi solusi terbaru dari solusi yang menggunakan resep User-Personalization. Kampanye yang dibuatnya secara otomatis diperbarui untuk menggunakan versi solusi masa depan. Kode menggunakan konfigurasi berikut:

- Ini mengonfigurasi kampanye untuk memperbarui secara otomatis untuk menggunakan versi solusi terbaru untuk solusi Anda: `solution-version-arn` Ini dalam *`solution ARN`*`/$LATEST` format, dan `syncWithLatestSolutionVersion` sedang`True`. Untuk menggunakan kode, ganti `solution ARN` dengan Amazon Resource Name (ARN) solusi Anda.

  Untuk menonaktifkan otomatis`syncWithLatestSolutionVersion`, tentukan hanya versi solusi ARN (tanpa`/$LATEST`), dan atur `syncWithLatestSolutionVersion` ke. `False`
- Ini menetapkan `enableMetadataWithRecommendations` opsi untuk`True`. Ini memungkinkan opsi permintaan rekomendasi untuk menyertakan metadata item dari kumpulan data Item dengan hasil rekomendasi. Untuk menonaktifkan opsi ini, atur ke`False`. Untuk informasi selengkapnya, lihat Metadata item dalam rekomendasi.
- Ini diatur `min-provisioned-tps` ke 1 (default). Sebaiknya mulai dengan 1 untuk `minProvisionedTPS` (default). Lacak penggunaan Anda dengan menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya. Untuk informasi selengkapnya, lihat Minimum transaksi yang disediakan per detik dan auto-scaling.

Untuk daftar lengkap semua parameter, lihatCreateCampaign.

```
aws personalize create-campaign \
--name campaign-name \
--solution-version-arn solution-arn/\$LATEST \
--min-provisioned-tps 1 \
--campaign-config "{"\"syncWithLatestSolutionVersion"\": "true",
 "\"enableMetadataWithRecommendations"\": "true"}"
```

Kampanye siap ketika statusnya AKTIF. Untuk mendapatkan status saat ini, panggil DescribeCampaign dan periksa apakah `status` bidangnya`ACTIVE`.

Jika Anda melatih ulang versi solusi dan kampanye Anda tidak diperbarui secara otomatis untuk menggunakan versi solusi terbaru, atau jika ingin mengubah setelan kampanye, Anda harus memperbarui kampanye. Untuk informasi selengkapnya, lihat Memperbarui konfigurasi kampanye Amazon Personalize.

Amazon Personalize memberi Anda operasi untuk mengelola kampanye seperti ListCampaigns mencantumkan kampanye yang telah Anda buat. Anda dapat menghapus kampanye dengan meneleponDeleteCampaign. Jika Anda menghapus kampanye, versi solusi yang merupakan bagian dari kampanye tidak akan dihapus.

Setelah Anda membuat kampanye, Anda dapat menggunakannya untuk membuat rekomendasi. Untuk informasi selengkapnya, lihat Mendapatkan rekomendasi dari Amazon Personalize.

Membuat kampanye (AWS SDKs)

> **Important**
>
> Anda dikenakan biaya kampanye saat kampanye aktif. Untuk menghindari biaya yang tidak perlu, pastikan untuk menghapus kampanye ketika Anda selesai. Untuk informasi tentang biaya kampanye, lihat harga Amazon Personalisasi.

Setelah versi solusi Anda Aktif, Anda siap untuk menerapkannya dengan kampanye Amazon Personalize. Untuk membuat kampanye dengan AWS SDKs, Anda menggunakan operasi CreateCampaign API.

Contoh kode berikut menunjukkan cara membuat kampanye. Kode ini menerapkan versi solusi terbaru dari solusi yang menggunakan resep User-Personalization. Kampanye yang dibuatnya secara otomatis diperbarui untuk menggunakan versi solusi masa depan. Kode menggunakan konfigurasi berikut:

- Ini mengonfigurasi kampanye untuk memperbarui secara otomatis untuk menggunakan versi solusi terbaru untuk solusi Anda: `solutionVersionArn` Ini dalam *`solution ARN`*`/$LATEST` format, dan `syncWithLatestSolutionVersion` sedang`True`. Untuk menggunakan kode, ganti `solution ARN` dengan Amazon Resource Name (ARN) versi solusi Anda.

  Untuk menonaktifkan otomatis`syncWithLatestSolutionVersion`, tentukan hanya versi solusi ARN (tanpa`/$LATEST`), dan atur `syncWithLatestSolutionVersion` ke. `False`
- Ini menetapkan `enableMetadataWithRecommendations` opsi untuk`True`. Ini memungkinkan opsi permintaan rekomendasi untuk menyertakan metadata item dari kumpulan data Item dengan hasil rekomendasi. Untuk menonaktifkan opsi ini, atur ke`False`. Untuk informasi selengkapnya, lihat Metadata item dalam rekomendasi.
- Ini diatur `minProvisionedTPS` ke 1 (default). Kami menyarankan Anda memulai dengan 1 untuk `minProvisionedTPS` (default). Lacak penggunaan Anda dengan menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya. Untuk informasi selengkapnya, lihat Minimum transaksi yang disediakan per detik dan auto-scaling.

Untuk daftar lengkap semua parameter, lihatCreateCampaign.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.create_campaign(
    name = 'campaign name',
    solutionVersionArn = 'solution ARN/$LATEST',
    minProvisionedTPS = 1,
    campaignConfig = {"syncWithLatestSolutionVersion": True,
 "enableMetadataWithRecommendations": True}
)

arn = response['campaignArn']

description = personalize.describe_campaign(campaignArn = arn)['campaign']
print('Name: ' + description['name'])
print('ARN: ' + description['campaignArn'])
print('Status: ' + description['status'])
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateCampaignCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({ region: "REGION" });

// set the campaign parameters
export const createCampaignParam = {
  solutionVersionArn: "SOLUTION_ARN/$LATEST" /* required */,
  name: "NAME" /* required */,
  minProvisionedTPS: 1 /* optional */,
  campaignConfig: {   /* optional */
    syncWithLatestSolutionVersion: true,
    enableMetadataWithRecommendations: true,
  },
};

export const run = async () => {
  try {
```

```
    const response = await personalizeClient.send(
      new CreateCampaignCommand(createCampaignParam)
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Kampanye siap ketika statusnya AKTIF. Untuk mendapatkan status saat ini, panggilDescribeCampaign, dan periksa apakah `status` bidangnya`ACTIVE`.

Jika Anda melatih ulang versi solusi secara manual, atau jika ingin mengubah setelan kampanye, Anda harus memperbarui kampanye. Untuk informasi selengkapnya, lihat Memperbarui konfigurasi kampanye Amazon Personalize.

Amazon Personalize memberi Anda operasi untuk mengelola kampanye seperti ListCampaigns mencantumkan kampanye yang telah Anda buat. Anda dapat menghapus kampanye dengan meneleponDeleteCampaign. Jika Anda menghapus kampanye, versi solusi yang merupakan bagian dari kampanye tidak akan dihapus.

Setelah Anda membuat kampanye, gunakan untuk membuat rekomendasi. Lihat informasi yang lebih lengkap di Mendapatkan rekomendasi dari Amazon Personalize.

Memperbarui konfigurasi kampanye Amazon Personalize

Untuk mengubah kampanye AndaTPS minimum yang disediakan, menerapkan versi solusi baru secara manual, atau mengubah konfigurasi kampanye, seperti mengaktifkan opsi untuk menyertakan metadata dalam rekomendasi, Anda harus memperbarui kampanye secara manual.

Berikut ini tidak memerlukan pembaruan kampanye manual:

- Jika kampanye Anda menggunakan pembaruan kampanye otomatis, Anda tidak perlu memperbaruinya untuk menerapkan versi solusi terbaru yang terlatih secara otomatis atau manual dari solusi Anda. Untuk informasi selengkapnya, lihat Pembaruan kampanye otomatis.
- Dengan User-Personalization-v 2, Personalisasi Pengguna, atau, Amazon Next-Best-Action Personalisasi secara otomatis memperbarui versi solusi terbaru Anda setiap dua jam untuk

menyertakan item atau tindakan baru dalam rekomendasi. Kampanye Anda secara otomatis menggunakan versi solusi yang diperbarui.

Anda memperbarui kampanye secara manual dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs

> **Note**
>
> Untuk memperbarui kampanye guna mengaktifkan opsi untuk menyertakan metadata dalam rekomendasi, Anda harus memiliki kumpulan data Item dengan kolom metadata. Anda tidak harus menggunakan metadata dalam pelatihan.
> Jika kampanye Anda sebelumnya menerapkan versi solusi yang menggunakan User-Personalization-v 2 atau Personalized-Ranking-v 2, dan Anda beralih ke versi resep yang lebih lama, opsi untuk menyertakan metadata dinonaktifkan secara default. Anda dapat mengaktifkannya saat memperbarui kampanye. Untuk informasi selengkapnya, lihat Metadata item dalam rekomendasi.

Topik

- Memperbarui kampanye (konsol)
- Memperbarui kampanye (AWS CLI)
- Memperbarui kampanye (AWS SDKs)

Memperbarui kampanye (konsol)

Untuk menerapkan versi solusi yang dilatih ulang secara manual atau membuat perubahan pada konfigurasi kampanye, Anda harus memperbarui kampanye.

Untuk memperbarui kampanye (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data dengan kampanye yang ingin Anda perbarui.
3. Di panel navigasi, pilih Kampanye.
4. Pada halaman Kampanye, pilih kampanye yang ingin Anda perbarui.
5. Pada halaman detail kampanye, pilih Perbarui.

6. Pada halaman Update campaign, buat perubahan. Misalnya, jika Anda menerapkan versi solusi yang dilatih ulang, untuk ID versi Solusi, pilih nomor identifikasi untuk versi solusi baru.
7. Pilih Perbarui. Amazon Personalize memperbarui kampanye untuk menggunakan versi solusi baru dan konfigurasi yang diubah.

Memperbarui kampanye (AWS CLI)

Untuk menerapkan versi solusi baru, mengubah kampanyeTPS minimum yang disediakan, atau mengubah konfigurasi kampanye, Anda harus memperbarui kampanye. Gunakan `update-campaign` perintah berikut untuk memperbarui kampanye agar menggunakan versi solusi baru dengan AWS CLI.

Ganti `campaign arn` dengan Nama Sumber Daya Amazon (ARN) dari kampanye yang ingin Anda perbarui. Ganti `new solution version arn` dengan versi solusi yang ingin Anda terapkan.

```
aws personalize update-campaign \
--campaign-arn campaign arn \
--solution-version-arn new solution version arn \
--min-provisioned-tps 1
```

Memperbarui kampanye (AWS SDKs)

Untuk menerapkan versi solusi baru, mengubah kampanye, TPS minimum yang disediakan atau mengubah konfigurasi kampanye, Anda harus memperbarui kampanye. Gunakan kode berikut untuk memperbarui kampanye dengan SDK for Python (Boto3) atau SDK for Java 2.x. Untuk daftar parameter lengkap, lihatUpdateCampaign.

SDK for Python (Boto3)

Gunakan `update_campaign` metode berikut untuk menerapkan versi solusi baru. Ganti `campaign arn` dengan Nama Sumber Daya Amazon (ARN) dari kampanye yang ingin Anda perbarui, ganti `new solution version arn` dengan versi solusi baru ARN dan secara opsional ubah. `minProvisionedTPS`

```
import boto3

personalize = boto3.client('personalize')

response = personalize.update_campaign(
```

```
    campaignArn = 'campaign arn',
    solutionVersionArn = 'new solution version arn',
    minProvisionedTPS = 1,
)

arn = response['campaignArn']

description = personalize.describe_campaign(campaignArn = arn)['campaign']
print('Name: ' + description['name'])
print('ARN: ' + description['campaignArn'])
print('Status: ' + description['status'])
```

SDK for Java 2.x

Gunakan `updateCampaign` metode berikut untuk memperbarui kampanye agar menggunakan versi solusi baru. Teruskan sebagai parameter klien layanan Amazon Personalize, Amazon Resource Name (ARN) versi solusi baru, dan file. TPS minimum yang disediakan

```
public static void updateCampaign(PersonalizeClient personalizeClient,
                                  String campaignArn,
                                  String solutionVersionArn,
                                  Integer minProvisionedTPS) {

    try {
        // build the updateCampaignRequest
        UpdateCampaignRequest updateCampaignRequest =
 UpdateCampaignRequest.builder()
            .campaignArn(campaignArn)
            .solutionVersionArn(solutionVersionArn)
            .minProvisionedTPS(minProvisionedTPS)
            .build();

        // update the campaign
        personalizeClient.updateCampaign(updateCampaignRequest);

        DescribeCampaignRequest campaignRequest = DescribeCampaignRequest.builder()
              .campaignArn(campaignArn)
              .build();

        DescribeCampaignResponse campaignResponse =
 personalizeClient.describeCampaign(campaignRequest);
        Campaign updatedCampaign = campaignResponse.campaign();
```

```
        System.out.println("The Campaign status is " + updatedCampaign.status());

    } catch (PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
}
```

Mendapatkan rekomendasi dari Amazon Personalize

Setelah Anda membuat pemberi rekomendasi atau membuat kampanye, Anda siap untuk mendapatkan rekomendasi. Bergantung pada sumber daya Anda, Anda bisa mendapatkan rekomendasi secara real time atau dengan alur kerja batch.

- Dengan sumber daya khusus, Anda bisa mendapatkan rekomendasi waktu nyata atau rekomendasi batch. Untuk rekomendasi real-time, Anda harus membuat kampanye khusus sebelum mendapatkan rekomendasi. Untuk rekomendasi batch, Anda tidak perlu membuat kampanye.
- Dengan pemberi rekomendasi dalam grup kumpulan data Domain, Anda hanya bisa mendapatkan rekomendasi waktu nyata.

Topik berikut menjelaskan bagaimana dan kapan menggunakan setiap jenis rekomendasi. Dengan rekomendasi batch dan real-time, Anda dapat memfilter hasil. Untuk mengetahui informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Topik

- Skor rekomendasi
- Rekomendasi item waktu nyata di Amazon Personalisasi
- Rekomendasi tindakan waktu nyata di Amazon Personalisasi
- Mendapatkan peringkat yang dipersonalisasi (sumber daya khusus)
- Meningkatkan relevansi rekomendasi dengan metadata kontekstual
- Mendapatkan rekomendasi item batch dengan sumber daya khusus
- Mendapatkan segmen pengguna batch dengan sumber daya khusus

Skor rekomendasi

Dengan solusi kustom yang dibuat dengan resep User-Personalization-v 2, User-Personalization, Personalized-Ranking-v 2, Personalized-Ranking, dan PERSONALIZED_ACTIONS, Amazon Personalize menyertakan skor untuk setiap item dalam rekomendasi. Skor ini mewakili kepastian relatif yang dimiliki Amazon Personalize tentang item atau tindakan mana yang akan dipilih pengguna selanjutnya. Skor yang lebih tinggi mewakili kepastian yang lebih besar.

- Untuk informasi tentang skor untuk User-Personalization-v 2 dan Personalisasi Pengguna, lihat. Cara kerja penilaian rekomendasi (sumber daya khusus)
- Untuk informasi tentang skor untuk resep PERSONALIZED_ACTIONS, lihat. Cara kerja penilaian rekomendasi tindakan
- Untuk informasi tentang skor untuk Personalized-Ranking-v 2 dan rekomendasi Peringkat yang Dipersonalisasi, lihat. Cara kerja penilaian peringkat yang dipersonalisasi

Untuk pekerjaan inferensi batch, skor item dihitung seperti yang dijelaskan dalam Cara kerja penilaian rekomendasi (sumber daya khusus) danCara kerja penilaian peringkat yang dipersonalisasi. Anda dapat melihat skor dalam file JSON keluaran pekerjaan inferensi batch.

Rekomendasi item waktu nyata di Amazon Personalisasi

Jika kasus penggunaan atau resep Anda menghasilkan rekomendasi item, setelah Anda membuat pemberi rekomendasi atau membuat kampanye, Anda bisa mendapatkan rekomendasi item yang dipersonalisasi atau terkait secara real-time untuk pengguna Anda.

Jika kasus penggunaan domain atau resep Anda menyediakan personalisasi real-time, seperti pilihan teratas untuk kasus penggunaan Anda atau resep User-personalisasi-v2, Amazon Personalize memperbarui rekomendasi berdasarkan aktivitas terbaru pengguna saat Anda merekam interaksinya dengan katalog Anda. Untuk informasi selengkapnya tentang merekam peristiwa dan personalisasi waktu nyata, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi.

Ketika Anda mendapatkan rekomendasi item real-time, Anda dapat melakukan hal berikut:

- Jika Anda mengonfigurasi kampanye untuk menampilkan metadata untuk item yang direkomendasikan, Anda dapat menentukan kolom yang akan disertakan dalam operasi GetRecommendations API Anda. Atau Anda dapat menentukan kolom saat menguji kampanye dengan konsol Amazon Personalize. Untuk contoh kode, lihatMendapatkan metadata item dengan rekomendasi waktu nyata. Untuk informasi tentang mengaktifkan metadata untuk kampanye, lihat. Metadata item dalam rekomendasi Untuk informasi tentang mengaktifkan metadata untuk pemberi rekomendasi, lihat. Mengaktifkan metadata dalam rekomendasi untuk pemberi rekomendasi domain di Amazon Personalize
- Untuk beberapa kasus penggunaan dan resep, Anda dapat menentukan promosi dalam permintaan rekomendasi Anda. Promosi mendefinisikan aturan bisnis tambahan yang berlaku untuk subset item yang direkomendasikan yang dapat dikonfigurasi. Untuk informasi selengkapnya, lihat Mempromosikan item dalam rekomendasi waktu nyata.

- Anda dapat memfilter hasil berdasarkan kriteria khusus. Misalnya, Anda mungkin tidak ingin merekomendasikan produk yang telah dibeli pengguna atau hanya merekomendasikan item untuk kelompok usia tertentu. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

> **Note**
>
> Jika Anda menggunakan resep kustom PERSONALIZED_RANKING, lihat. Mendapatkan peringkat yang dipersonalisasi (sumber daya khusus)

Topik

- Cara kerja penilaian rekomendasi (sumber daya khusus)
- Alasan rekomendasi dengan User-Personalization-v 2
- Mendapatkan rekomendasi item waktu nyata
- Mendapatkan metadata item dengan rekomendasi waktu nyata
- Mempromosikan item dalam rekomendasi waktu nyata

Cara kerja penilaian rekomendasi (sumber daya khusus)

Dengan resep User-Personalization-v 2 dan Personalisasi Pengguna, Amazon Personalize menghasilkan skor untuk item berdasarkan data interaksi dan metadata pengguna. Skor ini mewakili kepastian relatif yang dimiliki Amazon Personalize apakah pengguna akan berinteraksi dengan item berikutnya. Skor yang lebih tinggi mewakili kepastian yang lebih besar.

> **Note**
>
> Amazon Personalize tidak menampilkan skor untuk pemberi rekomendasi domain atau resep Similar-Item, SIMS, atau Hitung Popularitas. Untuk informasi tentang skor untuk rekomendasi Personalized-Ranking, lihat. Cara kerja penilaian peringkat yang dipersonalisasi

Amazon Personalize menghasilkan skor untuk item relatif satu sama lain dalam skala dari 0 hingga 1 (keduanya inklusif). Dengan User-Personalization-v 2, Amazon Personalize menghasilkan skor untuk subset item Anda. Dengan Personalisasi Pengguna, Amazon Personalize menilai semua item dalam katalog Anda.

Jika Anda menggunakan User-Personalization-v 2 dan menerapkan filter ke rekomendasi, tergantung pada berapa banyak rekomendasi yang dihapus filter, Amazon Personalize dapat menambahkan item placeholder. Hal ini dilakukan `numResults` untuk memenuhi permintaan rekomendasi Anda. Item ini adalah item populer, berdasarkan jumlah data interaksi, yang memenuhi kriteria filter Anda. Mereka tidak memiliki skor relevansi bagi pengguna.

Untuk User-Personalization-v 2 dan User-Personalization, total semua skor sama dengan 1. Misalnya, jika Anda mendapatkan rekomendasi film untuk pengguna dan ada tiga film yang muncul dalam kumpulan data Item dan kumpulan data Interaksi, skornya mungkin`0.6`,`0.3`, dan. `0.1` Demikian pula, jika Anda memiliki 10.000 film dalam inventaris Anda, film dengan skor tertinggi mungkin memiliki skor yang sangat kecil (skor rata-rata`.001`), tetapi, karena penilaian relatif, rekomendasinya masih berlaku.

Dalam istilah matematika, skor untuk setiap pasangan item pengguna (u, i) dihitung sesuai dengan rumus berikut, di mana fungsi eksponensial, w_u dan w_i/$_j$masing-masing `exp` adalah penyematan pengguna dan item, dan huruf Yunani sigma (Σ) mewakili penjumlahan atas semua item dengan skor:

$$\text{score}(u, i) = \frac{\exp(\bar{w}_u^\top w_i)}{\sum_j \exp(\bar{w}_u^\top w_j)}$$

Alasan rekomendasi dengan User-Personalization-v 2

Jika Anda menggunakan User-Personalization-v 2, item model biasanya tidak merekomendasikan menyertakan `reason` daftar. Alasan-alasan ini menjelaskan mengapa item tersebut dimasukkan dalam rekomendasi. Alasan yang mungkin termasuk yang berikut:

- Item promosi - Menunjukkan item tersebut disertakan sebagai bagian dari promosi yang Anda terapkan dalam permintaan rekomendasi Anda.
- Eksplorasi - Menunjukkan item itu disertakan dengan eksplorasi. Dengan eksplorasi, rekomendasi mencakup item dengan data interaksi yang lebih sedikit atau relevansi bagi pengguna. Untuk informasi lebih lanjut tentang eksplorasi, lihat Eksplorasi.
- Item populer - Menunjukkan item dimasukkan sebagai item populer placeholder. Jika Anda menggunakan filter, tergantung pada berapa banyak rekomendasi yang dihapus filter, Amazon Personalize dapat menambahkan item placeholder untuk memenuhi permintaan rekomendasi

Anda. `numResults` Item ini adalah item populer, berdasarkan data interaksi, yang memenuhi kriteria filter Anda. Mereka tidak memiliki skor relevansi bagi pengguna.

Mendapatkan rekomendasi item waktu nyata

Anda bisa mendapatkan rekomendasi item real-time dari pemberi rekomendasi Amazon Personalize atau kampanye khusus dengan konsol Amazon Personalize, (), AWS Command Line Interface atau.AWS CLI AWS SDKs

Topik

- Mendapatkan rekomendasi item (konsol)
- Mendapatkan rekomendasi item (AWS CLI)
- Mendapatkan rekomendasi item (AWS SDKs)

Mendapatkan rekomendasi item (konsol)

Untuk mendapatkan rekomendasi dengan konsol Amazon Personalize, Anda memberikan informasi permintaan di halaman detail pemberi rekomendasi (Grup kumpulan data Domain) atau kampanye khusus.

Untuk mendapatkan rekomendasi

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data yang berisi kampanye atau pemberi rekomendasi yang Anda gunakan.
3. Di panel navigasi, pilih Kampanye atau Rekomendasi.
4. Pilih kampanye target atau pemberi rekomendasi.
5. Untuk kampanye, di bawah Hasil kampanye uji, masukkan detail permintaan rekomendasi Anda berdasarkan resep yang Anda gunakan. Untuk pemberi rekomendasi pilih Test recommended dan masukkan rincian permintaan rekomendasi Anda berdasarkan kasus penggunaan Anda.

   Jika Anda merekam peristiwa untuk pengguna sebelum mereka masuk (pengguna anonim), Anda bisa mendapatkan rekomendasi untuk pengguna ini dengan memberikan `sessionId` dari peristiwa tersebut seolah-olah itu adalah milik merekaUserId. Untuk informasi selengkapnya tentang merekam peristiwa untuk pengguna anonim, lihatMerekam acara untuk pengguna anonim.

6. Secara opsional pilih filter. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.
7. Jika Anda menggunakan metadata kontekstual, berikan data untuk setiap konteks. Untuk setiap konteks, untuk Key masukkan bidang metadata. Untuk Nilai masukkan data konteks. Untuk informasi selengkapnya, lihat Meningkatkan relevansi rekomendasi dengan metadata kontekstual.
8. Jika Anda mengaktifkan metadata dalam rekomendasi untuk kampanye atau pemberi rekomendasi, untuk kolom kumpulan data Item, pilih kolom metadata yang ingin Anda sertakan dalam hasil rekomendasi. Untuk informasi tentang mengaktifkan metadata untuk kampanye, lihat. Metadata item dalam rekomendasi Untuk informasi tentang mengaktifkan metadata untuk pemberi rekomendasi, lihat. Mengaktifkan metadata dalam rekomendasi untuk pemberi rekomendasi domain di Amazon Personalize
9. Jika Anda ingin mempromosikan subset item, secara opsional lengkapi bidang Promosi. Untuk mengetahui informasi selengkapnya, lihat Mempromosikan item dalam rekomendasi waktu nyata.
10. Pilih Dapatkan rekomendasi. Tabel yang berisi 25 item direkomendasikan teratas pengguna ditampilkan. Jika Anda menggunakan User-Personalization-v 2, setiap item yang direkomendasikan mencakup daftar alasan mengapa item tersebut dimasukkan dalam rekomendasi. Untuk informasi selengkapnya, lihat Alasan rekomendasi dengan User-Personalization-v 2.

Mendapatkan rekomendasi item (AWS CLI)

Gunakan kode berikut untuk mendapatkan rekomendasi dari kampanye. Untuk mendapatkan rekomendasi dari pemberi rekomendasi, ganti `campaign-arn` parameter dengan. `recommender-arn`

Tentukan ID pengguna yang ingin Anda rekomendasikan, dan Nama Sumber Daya Amazon (ARN) kampanye atau pemberi rekomendasi Anda. Daftar 10 item yang direkomendasikan teratas untuk tampilan pengguna. Jika Anda menggunakan User-Personalization-v 2, setiap item yang direkomendasikan mencakup daftar alasan mengapa item tersebut dimasukkan dalam rekomendasi. Untuk informasi selengkapnya, lihat Alasan rekomendasi dengan User-Personalization-v 2.

Untuk mengubah jumlah item yang direkomendasikan, ubah nilainyanum`Results`. Defaultnya adalah 25 item. Maksimal adalah 500 item. Jika Anda menggunakan resep RELATED_ITEMS untuk melatih versi solusi yang mendukung kampanye, ganti `user-id` parameter dengan `item-id` dan tentukan ID item.

Jika Anda merekam peristiwa untuk pengguna sebelum mereka masuk (pengguna anonim), Anda bisa mendapatkan rekomendasi untuk pengguna ini dengan memberikan `sessionId` dari peristiwa tersebut seolah-olah itu adalah milik merekauserId. Untuk informasi selengkapnya tentang merekam peristiwa untuk pengguna anonim, lihatMerekam acara untuk pengguna anonim.

```
aws personalize-runtime get-recommendations \
--campaign-arn campaign arn \
--user-id User ID \
--num-results 10
```

Mendapatkan rekomendasi item (AWS SDKs)

Kode berikut menunjukkan cara mendapatkan rekomendasi Amazon Personalisasi untuk pengguna dari kampanye dengan. AWS SDKs Untuk mendapatkan rekomendasi dari pemberi rekomendasi, ganti `campaignArn` parameter dengan. `recommenderArn`

Tentukan ID pengguna yang ingin Anda rekomendasikan, dan Nama Sumber Daya Amazon (ARN) kampanye atau pemberi rekomendasi Anda. Daftar 10 item yang direkomendasikan teratas untuk tampilan pengguna. Jika Anda menggunakan User-Personalization-v 2, setiap item yang direkomendasikan mencakup daftar alasan mengapa item tersebut dimasukkan dalam rekomendasi. Untuk informasi selengkapnya, lihat Alasan rekomendasi dengan User-Personalization-v 2.

Untuk mengubah jumlah item yang direkomendasikan, ubah nilainyanumResults. Defaultnya adalah 25 item. Maksimal adalah 500 item. Jika Anda menggunakan resep RELATED_ITEMS untuk melatih versi solusi yang mendukung kampanye, ganti `userId` parameter dengan `itemId` dan tentukan ID item.

Jika Anda mengaktifkan metadata dalam rekomendasi untuk kampanye atau pemberi rekomendasi, Anda dapat menentukan kolom metadata kumpulan data Item yang akan disertakan dalam respons. Untuk contoh kode, lihatTermasuk metadata item dengan rekomendasi ()AWS SDKs. Untuk informasi tentang mengaktifkan metadata, lihat. Metadata item dalam rekomendasi

Jika Anda merekam peristiwa untuk pengguna sebelum mereka masuk (pengguna anonim), Anda bisa mendapatkan rekomendasi untuk pengguna ini dengan memberikan `sessionId` dari peristiwa tersebut seolah-olah itu adalah milik merekauserId. Untuk informasi selengkapnya tentang merekam peristiwa untuk pengguna anonim, lihatMerekam acara untuk pengguna anonim.

SDK for Python (Boto3)

```
import boto3
```

```
personalizeRt = boto3.client('personalize-runtime')

response = personalizeRt.get_recommendations(
    campaignArn = 'Campaign ARN',
    userId = 'User ID',
    numResults = 10
)

print("Recommended items")
for item in response['itemList']:
    print (item['itemId'])
```

SDK for Java 2.x

```
    public static void getRecs(PersonalizeRuntimeClient personalizeRuntimeClient,
 String campaignArn, String userId) {

        try {
            GetRecommendationsRequest recommendationsRequest =
 GetRecommendationsRequest.builder()
                    .campaignArn(campaignArn)
                    .numResults(20)
                    .userId(userId)
                    .build();

            GetRecommendationsResponse recommendationsResponse =
 personalizeRuntimeClient
                    .getRecommendations(recommendationsRequest);
            List<PredictedItem> items = recommendationsResponse.itemList();
            for (PredictedItem item : items) {
                System.out.println("Item Id is : " + item.itemId());
                System.out.println("Item score is : " + item.score());
            }

        } catch (AwsServiceException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { GetRecommendationsCommand } from "@aws-sdk/client-personalize-runtime";

import { personalizeRuntimeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region:
 "REGION"});

// Set the recommendation request parameters.
export const getRecommendationsParam = {
  campaignArn: "CAMPAIGN_ARN" /* required */,
  userId: "USER_ID" /* required */,
  numResults: 15 /* optional */,
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(
      new GetRecommendationsCommand(getRecommendationsParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Mendapatkan metadata item dengan rekomendasi waktu nyata

Jika Anda mengonfigurasi kampanye atau pemberi rekomendasi untuk menampilkan metadata untuk item yang direkomendasikan, Anda dapat menentukan kolom yang akan disertakan dalam operasi API AndaGetRecommendations. Atau Anda dapat menentukan kolom saat menguji kampanye dengan konsol Amazon Personalize.

Untuk informasi tentang mengaktifkan metadata untuk kampanye, lihat. Metadata item dalam rekomendasi Untuk informasi tentang mengaktifkan metadata untuk pemberi rekomendasi, lihat.

Mengaktifkan metadata dalam rekomendasi untuk pemberi rekomendasi domain di Amazon Personalize

Contoh kode berikut menunjukkan cara menentukan kolom metadata untuk disertakan dengan AWS CLI atau. AWS SDKs Untuk melakukannya dengan konsol Amazon Personalize, Anda menentukan kolom saat menguji kampanye. Untuk informasi selengkapnya, lihat Mendapatkan rekomendasi item (konsol).

Topik

- Termasuk metadata item dengan rekomendasi ()AWS CLI
- Termasuk metadata item dengan rekomendasi ()AWS SDKs

Termasuk metadata item dengan rekomendasi ()AWS CLI

Jika Anda mengaktifkan metadata dalam rekomendasi untuk kampanye atau pemberi rekomendasi, Anda dapat menentukan kolom metadata kumpulan data Item yang akan disertakan dalam respons. Contoh kode berikut menunjukkan cara menentukan kolom metadata sebagai bagian dari permintaan rekomendasi Anda.

```
aws personalize-runtime get-recommendations \
--campaign-arn campaign arn \
--user-id User ID \
--num-results 10 \
--metadata-columns "{\"ITEMS\": [\"\"columnNameA\"\",\"\"columnNameB\"\"]}"
```

Termasuk metadata item dengan rekomendasi ()AWS SDKs

Jika Anda mengaktifkan metadata dalam rekomendasi untuk kampanye atau pemberi rekomendasi, Anda dapat menentukan kolom metadata kumpulan data Item yang akan disertakan dalam respons. Contoh kode berikut menunjukkan cara menentukan kolom metadata sebagai bagian dari permintaan rekomendasi Anda.

```
import boto3

personalizeRt = boto3.client('personalize-runtime')

response = personalizeRt.get_recommendations(
  campaignArn = 'Campaign ARN',
  userId = 'User ID',
```

```
  numResults = 10
  metadataColumns = {
    "ITEMS": ['columnNameA','columnNameB']
  }
)

print("Recommended items")
for item in response['itemList']:
  print(item['itemId'])
  print(item['metadata'])
```

Mempromosikan item dalam rekomendasi waktu nyata

Dengan semua kasus penggunaan domain dan beberapa resep khusus, Anda dapat menentukan promosi saat Anda mendapatkan rekomendasi waktu nyata.

Promosi mendefinisikan aturan bisnis tambahan yang berlaku untuk subset item yang direkomendasikan yang dapat dikonfigurasi. Misalnya, Anda mungkin memiliki aplikasi streaming dan ingin mempromosikan acara dan film Anda sendiri, tetapi juga merekomendasikan judul yang relevan. Anda dapat menggunakan promosi untuk menentukan bahwa persentase tertentu dari item yang direkomendasikan harus berasal dari kategori in-house. Item yang direkomendasikan yang tersisa akan terus menjadi rekomendasi yang relevan berdasarkan resep Anda dan filter permintaan apa pun.

Untuk menerapkan promosi, Anda menentukan hal berikut dalam permintaan rekomendasi Anda:

- Persentase item yang direkomendasikan untuk menerapkan filter promosi.
- Filter yang menentukan kriteria promosi. Untuk informasi selengkapnya, lihat Filter promosi.

Dalam respons rekomendasi, item yang dipromosikan diposisikan secara acak relatif terhadap item lain yang direkomendasikan, tetapi dalam urutan yang diurutkan relatif terhadap item promosi lainnya. Tergantung pada resep Anda, item yang direkomendasikan yang bukan bagian dari promosi diurutkan berdasarkan relevansi dengan pengguna, popularitas, atau kesamaan. Jika tidak ada cukup item yang memenuhi kriteria promosi, hasilnya akan berisi sebanyak mungkin item yang dipromosikan.

Anda dapat menerapkan promosi ke rekomendasi dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs

Topik

- Gunakan kasing dan resep yang mendukung promosi
- Filter promosi
- Mempromosikan item baru
- Mempromosikan item (konsol)
- Mempromosikan item (AWS CLI)
- Mempromosikan item (AWS SDKs)

Gunakan kasing dan resep yang mendukung promosi

Semua kasus penggunaan mendukung promosi. Resep khusus berikut mendukung promosi:

- Personalisasi pengguna-v2dan Personalisasi Pengguna resep
- Barang Serupadan SIMS resep
- Trend-Sekarangdan Hitungan Popularitas resep

Filter promosi

Saat Anda menerapkan promosi ke permintaan rekomendasi, Anda memilih filter yang menentukan kriteria promosi. Anda dapat menggunakan filter yang ada atau membuat yang baru. Anda membuat dan mengelola filter untuk promosi seperti halnya filter lain di Amazon Personalisasi. Untuk informasi tentang membuat dan mengelola filter, lihatHasil penyaringan.

Satu-satunya perbedaan antara filter promosi dan filter yang Anda pilih di luar promosi (filter permintaan) adalah bagaimana Amazon Personalize menerapkannya. Filter promosi hanya berlaku untuk item yang dipromosikan, sedangkan filter permintaan hanya berlaku untuk item yang direkomendasikan yang tersisa. Jika Anda menentukan filter permintaan dan filter promosi, dan ingin menerapkan kedua filter ke item yang dipromosikan, ekspresi filter promosi Anda harus menyertakan kedua ekspresi tersebut. Cara Anda menggabungkan dua ekspresi tergantung pada kumpulan data yang Anda gunakan. Untuk informasi selengkapnya tentang ekspresi filter, aturannya, dan cara membuatnya, lihatEkspresi Filter.

Contoh ekspresi filter

Ekspresi berikut hanya mencakup item dari kategori “in-house”. Anda dapat menggunakan ungkapan ini jika Anda ingin mempromosikan konten Anda sendiri dalam rekomendasi Anda.

```
INCLUDE ItemID WHERE Items.OWNER IN ("in-house")
```

Ekspresi berikut hanya mencakup item yang dibuat lebih baru daripada stempel waktu yang Anda tentukan. Anda dapat menggunakan ungkapan ini untuk mempromosikan item baru dalam rekomendasi.

```
INCLUDE ItemID WHERE Items.CREATION_TIMESTAMP > $DATE
```

Ekspresi berikut menunjukkan bagaimana Anda dapat menerapkan filter permintaan ke item yang dipromosikan. Ini hanya mencakup item pakaian yang tersedia sebagai item yang dipromosikan. Dalam skenario ini, juga `Items.AVAILABLE IN ("True")` akan digunakan dalam ekspresi filter permintaan, sehingga semua rekomendasi adalah untuk item yang tersedia.

```
INCLUDE ItemID WHERE Items.CATEGORY IN ("clothing") AND Items.AVAILABLE IN ("True")
```

Untuk daftar contoh filter yang lebih lengkap, lihatContoh ekspresi filter.

Mempromosikan item baru

Jika Anda menggunakanResep personalisasi-v2 pengguna, Amazon Personalize merekomendasikan item yang paling relevan kepada pengguna dan lebih sering merekomendasikan item yang ada dengan data interaksi. Untuk memastikan rekomendasi menyertakan beberapa item baru, Anda dapat menerapkan promosi ke permintaan rekomendasi yang menyertakan item berdasarkan stempel waktu pembuatan.

Jika Anda belum menggunakan promosi, ekspresi filter Anda dapat mempromosikan item yang dibuat setelah tanggal tertentu:

```
INCLUDE ItemID WHERE Items.CREATION_TIMESTAMP > $DATE
```

Jika Anda sudah menggunakan promosi, Anda membuat ekspresi yang merantai promosi dan pernyataan kondisi item baru:

```
INCLUDE ItemID WHERE Items.CATEGORY IN ("clothing") OR Items.CREATION_TIMESTAMP > $DATE
```

Mempromosikan item (konsol)

Untuk mempromosikan item tertentu dalam rekomendasi dengan konsol Amazon Personalize, buat filter, lalu berikan detail promosi dalam permintaan rekomendasi. Untuk informasi tentang bidang lain, lihatMendapatkan rekomendasi item (konsol).

Untuk mempromosikan item dalam rekomendasi

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data yang berisi kampanye atau pemberi rekomendasi yang Anda gunakan.
3. Jika Anda belum melakukannya, buat filter yang menentukan kriteria promosi. Anda membuat filter untuk promosi dengan cara yang sama seperti Anda membuat filter permintaan. Untuk informasi tentang membuat dan mengelola filter, lihatHasil penyaringan.
4. Di panel navigasi, pilih Rekomendasi atau Kampanye.
5. Pilih kampanye target atau pemberi rekomendasi.
6. Untuk kampanye, di bawah Hasil kampanye Uji, masukkan detail permintaan rekomendasi Anda berdasarkan resep yang Anda gunakan. Untuk pemberi rekomendasi, pilih Rekomendasi tes dan masukkan detail permintaan rekomendasi Anda.
7. Secara opsional pilih filter untuk permintaan. Filter ini hanya berlaku untuk item yang tidak dipromosikan. Untuk informasi tentang membuat dan mengelola filter, lihatHasil penyaringan.
8. Jika Anda menggunakan metadata kontekstual, berikan data untuk setiap konteks. Untuk setiap konteks, untuk Key masukkan bidang metadata. Untuk Nilai, masukkan data konteks. Untuk informasi selengkapnya, lihat Meningkatkan relevansi rekomendasi dengan metadata kontekstual.
9. Untuk Promosi tentukan hal berikut:

   - Persen item yang dipromosikan: Masukkan persentase item yang direkomendasikan untuk menerapkan promosi.
   - Filter: Pilih filter yang menentukan kriteria promosi. Filter ini berlaku untuk item yang dipromosikan, bukan filter permintaan apa pun yang mungkin telah Anda tentukan di langkah 7.
   - Parameter filter: Jika promosi Anda menggunakan filter dengan parameter placeholder, untuk setiap parameter, masukkan nilai untuk menetapkan kriteria filter. Untuk menggunakan beberapa nilai untuk satu parameter, pisahkan setiap nilai dengan koma.
10. Pilih Dapatkan rekomendasi. Tabel yang berisi 25 item direkomendasikan teratas pengguna ditampilkan. Kolom item Promosi menunjukkan apakah item tersebut disertakan karena promosi Anda. Item yang dipromosikan diposisikan secara acak relatif terhadap item lain yang direkomendasikan, tetapi dalam urutan yang diurutkan relatif terhadap item promosi lainnya. Bergantung pada kasus penggunaan atau resep Anda, item yang direkomendasikan yang bukan bagian dari promosi diurutkan berdasarkan relevansi dengan pengguna, popularitas,

atau kesamaan. Jika tidak ada cukup item yang memenuhi kriteria promosi, hasilnya akan berisi sebanyak mungkin item yang dipromosikan.

Mempromosikan item (AWS CLI)

Kode berikut menunjukkan cara mempromosikan item dalam rekomendasi dengan AWS CLI dan kampanye khusus. Untuk mempromosikan item dengan pemberi rekomendasi, ganti `campaign-arn` parameter dengan `recommender-arn` dan tentukan Nama Sumber Daya Amazon (ARN) untuk pemberi rekomendasi. Untuk bidang promosi, tentukan yang berikut ini:

- nama: Beri nama promosi. Respons rekomendasi menggunakan nama untuk mengidentifikasi item yang dipromosikan.
- percent-promoted-items: Persentase item yang direkomendasikan untuk menerapkan promosi. Dalam contoh ini, 50% item akan dipromosikan item.
- FilterARN: Tentukan Nama Sumber Daya Amazon (ARN) dari filter yang menentukan kriteria promosi. Untuk informasi selengkapnya, lihat Filter promosi.
- nama dan nilai parameter: Jika ekspresi filter Anda memiliki parameter apa pun, berikan nama parameter (peka huruf besar/kecil) dan nilainya. Misalnya, jika ekspresi filter Anda memiliki `$GENRE` parameter, berikan GENRE sebagai kunci, dan genre atau genre, seperti Komedi, sebagai nilai. Pisahkan beberapa nilai dengan koma. Saat Anda menggunakan AWS CLI, untuk setiap nilai Anda harus menggunakan ` \ ` karakter untuk menghindari tanda kutip dan ` \ ` karakter. Contoh kode berikut menunjukkan bagaimana memformat nilai-nilai.

Kode menunjukkan cara menggunakan filter permintaan dan filter promosi. Filter promosi hanya berlaku untuk item yang dipromosikan, sedangkan filter permintaan hanya berlaku untuk item yang direkomendasikan yang tersisa. Untuk informasi selengkapnya, lihat Filter promosi.

Untuk informasi tentang bidang tambahan, lihat Mendapatkan rekomendasi item (AWS SDKs) danMendapatkan peringkat yang dipersonalisasi menggunakan metadata kontekstual.

```
aws personalize-runtime get-recommendations \
--campaign-arn CampaignArn \
--user-id 1 \
--num-results 10  \
--filter-arn RequestFilterArn \
--filter-values '{
    "RequestFilterParameterName": "\"value\"",
```

```
    "RequestFilterParameterName": "\"value1\",\"value2\",\"value3\""
  }' \
--promotions "[{
  \"name\": \"promotionName\",
  \"percentPromotedItems\": 50,
  \"filterArn\": \"PromotionFilterARN\",
  \"filterValues\": {\"PromotionParameterName\":\"\\\"value1, value2\\\"\"}
}]"
```

Daftar item yang direkomendasikan ditampilkan. Item yang dipromosikan diposisikan secara acak relatif terhadap item lain yang direkomendasikan, tetapi dalam urutan yang diurutkan relatif terhadap item promosi lainnya. Tergantung pada resep Anda, item yang direkomendasikan yang bukan bagian dari promosi diurutkan berdasarkan relevansi dengan pengguna, popularitas, atau kesamaan. Jika tidak ada cukup item yang memenuhi kriteria promosi, hasilnya akan berisi sebanyak mungkin item yang dipromosikan.

```
{
  "itemList": [
      {
          "itemId1": "123",
          "score": .0117211,
          "promotionName": "promotionName"
      },
      {
         "itemId2": "456",
         "score": .0077976
      },
      {
         "itemId3": "789",
         "score": .0067171
      },
      .....
 ]
```

Mempromosikan item (AWS SDKs)

Kode berikut menunjukkan cara mempromosikan item dalam rekomendasi dengan SDK for Python (Boto3) dan SDK for Java 2.x dan kampanye kustom. Untuk mempromosikan item dengan pemberi rekomendasi, ganti `campaignArn` parameter dengan `recommenderArn` dan tentukan Nama Sumber Daya Amazon (ARN) untuk pemberi rekomendasi. Untuk bidang promosi, tentukan yang berikut ini:

- nama: Tentukan nama promosi. Tanggapan rekomendasi mencakup nama untuk mengidentifikasi item yang dipromosikan.
- percentPromotedItems: Persentase item yang direkomendasikan untuk menerapkan promosi.
- PromotionFilterARN: Nama Sumber Daya Amazon (ARN) dari filter yang menentukan kriteria promosi. Untuk informasi selengkapnya, lihat Filter promosi.
- Nama dan nilai parameter apa pun: Jika ekspresi filter Anda memiliki parameter apa pun, untuk setiap parameter dalam ekspresi filter Anda, berikan nama parameter (peka huruf besar/kecil) dan nilainya. Misalnya, jika ekspresi filter Anda memiliki `$GENRE` parameter, berikan `"GENRE"` sebagai kunci, dan genre atau genre, seperti “\" Comedy "\”, sebagai nilai. Pisahkan beberapa nilai dengan koma. Misalnya, `"\"comedy\",\"drama\",\"horror"\"`.

Kode berikut menunjukkan cara menggunakan filter permintaan dan filter promosi. Filter promosi hanya berlaku untuk item yang dipromosikan, sedangkan filter permintaan hanya berlaku untuk item yang direkomendasikan yang tersisa. Untuk informasi selengkapnya, lihat Filter promosi.

Untuk informasi tentang bidang tambahan, lihat Mendapatkan rekomendasi item (AWS SDKs) danMendapatkan peringkat yang dipersonalisasi menggunakan metadata kontekstual.

SDK for Python (Boto3)

```
import boto3

personalizeRt = boto3.client('personalize-runtime')

response = personalizeRt.get_recommendations(
  campaignArn = "CampaignARN",
  userId = '1',
  numResults = 10,
  filterArn = 'RequestFilterARN',
  filterValues = {
      "RequestFilterParameterName": "\"value1\"",
      "RequestFilterParameterName": "\"value1\",\"value2\",\"value3\""
      ....
  },
  promotions = [{
    "name" : "promotionName",
    "percentPromotedItems" : 50,
    "filterArn": "promotionFilterARN",
    "filterValues": {
      "PromotionParameterName": "\"Value1\",\"Value2\""
```

```
      ...
    }
  }]
)

print("Recommended items")
for item in response['itemList']:
    print (item['itemId'])
    if ("promotionName" in item):
        print(item['promotionName'])
```

SDK for Java 2.x

```
public static void getRecommendationsWithPromotedItems(PersonalizeRuntimeClient
 personalizeRuntimeClient,
                                        String campaignArn,
                                        String userId,
                                        String requestFilterArn,
                                        String requestParameterName,
                                        String requestParameterValue1,
                                        String requestParameterValue2,
                                        String promotionName,
                                        int percentPromotedItems,
                                        String promotionFilterArn,
                                        String promotionParameterName,
                                        String promotionParameterValue1,
                                        String promotionParameterValue2) {

  try {

      Map<String, String> promotionFilterValues = new HashMap<>();

      promotionFilterValues.put(promotionParameterName, String.format("\"%1$s\",
\"%2$s\"",
              promotionParameterValue1, promotionParameterValue2));

      Promotion newPromotion = Promotion.builder()
              .name(promotionName)
              .percentPromotedItems(percentPromotedItems)
              .filterArn(promotionFilterArn)
              .filterValues(promotionFilterValues)
              .build();
```

```
      List<Promotion> promotionList = new List<>();

      promotionList.add(newPromotion);

      Map<String, String> requestfilterValues = new HashMap<>();

      requestfilterValues.put(requestParameterName, String.format("\"%1$s\",\"%2$s
\"",
              requestParameterValue1, requestParameterValue2));

      GetRecommendationsRequest recommendationsRequest =
 GetRecommendationsRequest.builder()
              .campaignArn(campaignArn)
              .numResults(20)
              .userId(userId)
              .filterArn(requestFilterArn)
              .fitlerValues(requestFilterValues)
              .promotions(promotionList)
              .build();

      GetRecommendationsResponse recommendationsResponse =
 personalizeRuntimeClient.getRecommendations(recommendationsRequest);
      List<PredictedItem> items = recommendationsResponse.itemList();

      for (PredictedItem item: items) {
          System.out.println("Item Id is : "+item.itemId());
          System.out.println("Item score is : "+item.score());
          System.out.println("Promotion name is : "+item.promotionName());
      }
  } catch (PersonalizeRuntimeException e) {
      System.err.println(e.awsErrorDetails().errorMessage());
      System.exit(1);
  }
}
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { GetRecommendationsCommand, PersonalizeRuntimeClient } from
  "@aws-sdk/client-personalize-runtime";

// create personalizeRuntimeClient.
const personalizeRuntimeClient = new PersonalizeRuntimeClient({
```

```
  region: "REGION",
});

// set recommendation request param
export const getRecommendationsParam = {
  campaignArn: "CAMPAIGN_ARN",  /* required */
  userId: "USER_ID", /* required */
  numResults: 25, /* optional */
  filterArn: "FILTER_ARN", /* provide if you are applying a custom filter */
  filterValues: {
    "PARAM_NAME": "\"PARAM_VALUE\"" /* provide if your filter has a placeholder
 parameter */
  },
  promotions: [
    {
      name: "PROMOTION_NAME", /* specify the name of the promotion. The
 recommendation response includes the name to identify promoted items. */
      percentPromotedItems: 50, /* the percentage of recommended items to apply the
 promotion to. */
      filterArn:
        "PROMOTION_FILTER_ARN", /* the Amazon Resource Name (ARN) of the filter that
 defines the promotion criteria. */
      filterValues: {
        "PARAM_NAME": "\"PARAM_VALUE\"" /* provide if your promotion filter has a
 placeholder parameter */
      },
    },
  ],
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(new
 GetRecommendationsCommand(getRecommendationsParam));
    console.log("Success!", "\nItems are: ");
    response.itemList.forEach(element => console.log(element.itemId))
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Daftar item yang direkomendasikan ditampilkan. Item yang dipromosikan diposisikan secara acak relatif terhadap item lain yang direkomendasikan, tetapi dalam urutan yang diurutkan relatif terhadap item promosi lainnya. Tergantung pada resep Anda, item yang direkomendasikan yang bukan bagian dari promosi diurutkan berdasarkan relevansi dengan pengguna, popularitas, atau kesamaan. Jika tidak ada cukup item yang memenuhi kriteria promosi, hasilnya akan berisi sebanyak mungkin item yang dipromosikan.

```
{
  "itemList": [
      {
          "itemId1": "123",
          "score": .0117211,
          "promotionName": "promotionName"
      },
      {
         "itemId2": "456",
         "score": .0077976
      },
      {
         "itemId3": "789",
         "score": .0067171
      },
      .....
 ]
```

Rekomendasi tindakan waktu nyata di Amazon Personalisasi

Jika Anda menggunakan resep PERSONALIZED_ACTIONS, Anda bisa mendapatkan rekomendasi tindakan dari kampanye Anda secara real time. Anda bisa mendapatkan rekomendasi tindakan dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs

Topik

- Cara kerja penilaian rekomendasi tindakan
- Mendapatkan rekomendasi tindakan (konsol)
- Mendapatkan rekomendasi tindakan (AWS CLI)
- Mendapatkan rekomendasi tindakan (AWS SDKs)

Cara kerja penilaian rekomendasi tindakan

Dengan Next-Best-Action resepnya, Amazon Personalize menghasilkan skor untuk tindakan berdasarkan kemungkinan pengguna akan berinteraksi dengan tindakan tersebut. Skor bisa antara 0 - 1.0. Semakin dekat ke 1.0, semakin besar kemungkinan pengguna akan berinteraksi dengan tindakan tersebut.

Jika Anda belum mengimpor data interaksi tindakan apa pun, semua tindakan yang disarankan akan memiliki skor 0,0. Jika Amazon Personalize merekomendasikan tindakan sebagai bagian dari eksplorasi, item tersebut akan memiliki skor 0,0. Amazon Personalize menggunakan eksplorasi untuk merekomendasikan tindakan tanpa data interaksi tindakan. Untuk informasi lebih lanjut tentang eksplorasi, lihatEksplorasi.

Mendapatkan rekomendasi tindakan (konsol)

Untuk mendapatkan rekomendasi tindakan dengan konsol Amazon Personalize, Anda memberikan informasi permintaan di halaman detail kampanye kustom Anda.

Untuk mendapatkan rekomendasi tindakan

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data yang berisi kampanye yang Anda gunakan.
3. Di panel navigasi, di bawah Sumber daya khusus, pilih Kampanye.
4. Pilih kampanye target.
5. Di bawah Uji hasil kampanye, masukkan detail permintaan rekomendasi Anda.

   Jika Anda merekam peristiwa untuk pengguna sebelum mereka masuk (pengguna anonim), Anda bisa mendapatkan rekomendasi untuk pengguna ini dengan memberikan `sessionId` dari peristiwa tersebut seolah-olah itu adalah milik merekaUserId. Untuk informasi selengkapnya tentang merekam peristiwa untuk pengguna anonim, lihatMerekam acara untuk pengguna anonim.
6. Secara opsional pilih filter. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.
7. Pilih Dapatkan rekomendasi. Tabel yang berisi 5 tindakan direkomendasikan teratas pengguna muncul.

Mendapatkan rekomendasi tindakan (AWS CLI)

Gunakan kode berikut untuk mendapatkan rekomendasi tindakan dari kampanye. Tentukan ID pengguna yang ingin Anda rekomendasikan dan Nama Sumber Daya Amazon (ARN) kampanye Anda.

Untuk mengubah jumlah tindakan yang disarankan, ubah nilainya untuk`numResults`. Defaultnya adalah 5 tindakan. Maksimal adalah 100 tindakan.

Untuk memfilter rekomendasi tindakan berdasarkan kriteria khusus, Anda dapat membuat filter dan menerapkannya pada `get-action-recommendations` operasi. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Jika Anda merekam peristiwa untuk pengguna sebelum mereka masuk (pengguna anonim), Anda bisa mendapatkan rekomendasi untuk pengguna ini dengan memberikan `sessionId` dari peristiwa tersebut seolah-olah itu adalah milik merekа`userId`. Untuk informasi selengkapnya tentang merekam peristiwa untuk pengguna anonim, lihatMerekam acara untuk pengguna anonim.

```
aws personalize-runtime get-action-recommendations \
--campaign-arn campaign arn \
--user-id User ID \
--num-results 10
```

Mendapatkan rekomendasi tindakan (AWS SDKs)

Kode berikut menunjukkan cara mendapatkan rekomendasi Amazon Personalisasi untuk pengguna dari kampanye. Tentukan ID pengguna yang ingin Anda rekomendasikan, dan Nama Sumber Daya Amazon (ARN) kampanye Anda.

Untuk mengubah jumlah tindakan yang disarankan, ubah nilainya untuk`numResults`. Defaultnya adalah 5 tindakan. Maksimal adalah 100 tindakan.

Untuk memfilter rekomendasi tindakan berdasarkan kriteria khusus, Anda dapat membuat filter dan menerapkannya ke permintaan GetActionRecommendations API. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Jika Anda merekam peristiwa untuk pengguna sebelum mereka masuk (pengguna anonim), Anda bisa mendapatkan rekomendasi untuk pengguna ini dengan memberikan `sessionId` dari peristiwa tersebut seolah-olah itu adalah milik merekа`userId`. Untuk informasi selengkapnya tentang merekam peristiwa untuk pengguna anonim, lihatMerekam acara untuk pengguna anonim.

```
import boto3

personalizeRt = boto3.client('personalize-runtime')

response = personalizeRt.get_action_recommendations(
    campaignArn = 'Campaign ARN',
    userId = 'User ID',
    numResults = 10
)

print("Recommended actions")
for item in response['actionList']:
    print (item['actionId'])
```

Mendapatkan peringkat yang dipersonalisasi (sumber daya khusus)

Peringkat yang dipersonalisasi adalah daftar item yang direkomendasikan yang diberi peringkat ulang untuk pengguna tertentu. Untuk mendapatkan peringkat yang dipersonalisasi, hubungi operasi GetPersonalizedRanking API atau dapatkan rekomendasi dari kampanye di konsol.

> **Note**
>
> Solusi yang mendukung kampanye harus dibuat menggunakan resep tipe PERSONALIZED_RANKING. Untuk informasi selengkapnya, lihat Memilih resep.

Topik

- Cara kerja penilaian peringkat yang dipersonalisasi
- Mendapatkan peringkat yang dipersonalisasi (konsol)
- Mendapatkan peringkat yang dipersonalisasi (AWS CLI)
- Mendapatkan peringkat yang dipersonalisasi (AWS SDKs)
- Notebook sampel Peringkat yang Dipersonalisasi

Cara kerja penilaian peringkat yang dipersonalisasi

Seperti skor yang dikembalikan oleh `GetRecommendations` operasi untuk solusi yang dibuat dengan resep User-Personalization-v 2 dan Personalisasi Pengguna, `GetPersonalizedRanking`

skor berjumlah 1, tetapi hanya item input yang menerima skor dan skor rekomendasi cenderung lebih tinggi. Jika item tidak hadir selama pelatihan terbaru, ia menerima skor 0.

Secara matematis, fungsi penilaian untuk GetPersonalizedRanking identik dengan`GetRecommendations`, kecuali bahwa ia hanya mempertimbangkan item input. Ini berarti bahwa skor mendekati 1 menjadi lebih mungkin, karena ada lebih sedikit pilihan lain untuk membagi skor:

$$\text{score}(u,i) = \frac{\exp(\bar{w}_u^\top w_i)}{\sum_{j \in \text{input}} \exp(\bar{w}_u^\top w_j)}$$

Mendapatkan peringkat yang dipersonalisasi (konsol)

Untuk mendapatkan peringkat yang dipersonalisasi untuk pengguna dari konsol Amazon Personalize, pilih kampanye yang Anda gunakan, lalu berikan ID pengguna mereka, tentukan daftar item yang ingin diberi peringkat untuk pengguna, pilih filter secara opsional, dan berikan data konteks apa pun secara opsional.

Untuk mendapatkan peringkat yang dipersonalisasi untuk pengguna

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data yang berisi kampanye yang Anda gunakan.
3. Di panel navigasi, pilih Kampanye.
4. Pada halaman Kampanye, pilih kampanye target.
5. Di bawah Uji hasil kampanye, masukkan ID Pengguna pengguna yang ingin Anda rekomendasikan.
6. Untuk Item IDs, masukkan daftar item yang akan diberi peringkat bagi pengguna.
7. Secara opsional pilih filter. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.
8. Jika Anda mengaktifkan metadata dalam rekomendasi untuk kampanye Anda, untuk kolom kumpulan data Item, pilih kolom metadata yang ingin Anda sertakan dalam hasil rekomendasi. Untuk informasi tentang mengaktifkan metadata, lihat. Metadata item dalam rekomendasi

9. Jika kampanye Anda menggunakan metadata kontekstual (untuk persyaratan lihatMeningkatkan relevansi rekomendasi dengan metadata kontekstual) secara opsional, berikan data konteks.

   Untuk setiap konteks, untuk Kunci, masukkan bidang metadata, dan untuk Nilai, masukkan data konteks.

10. Pilih Dapatkan peringkat item yang dipersonalisasi. Tabel yang berisi item yang diberi peringkat dalam urutan minat yang diprediksi untuk pengguna muncul.

Mendapatkan peringkat yang dipersonalisasi (AWS CLI)

Contoh kode berikut menunjukkan bagaimana variasi yang berbeda tentang cara mendapatkan peringkat yang dipersonalisasi dengan AWS CLI.

Topik

- Mendapatkan peringkat yang dipersonalisasi
- Termasuk metadata item dalam peringkat yang dipersonalisasi

Mendapatkan peringkat yang dipersonalisasi

Gunakan `get-personalized-ranking` perintah berikut untuk mendapatkan peringkat yang dipersonalisasi dengan AWS CLI. Tentukan Nama Sumber Daya Amazon (ARN) untuk kampanye Anda, ID Pengguna untuk pengguna, dan berikan daftar item IDs untuk item yang akan diberi peringkat bagi pengguna (masing-masing dipisahkan oleh spasi). Item yang akan diberi peringkat harus dalam data yang Anda gunakan untuk melatih versi solusi. Daftar rekomendasi peringkat ditampilkan. Amazon Personalize mempertimbangkan item pertama dalam daftar yang paling menarik bagi pengguna.

```
aws personalize-runtime get-personalized-ranking \
--campaign-arn Campaign ARN \
--user-id 12 \
--input-list 3 4 10 8 12 7
```

Termasuk metadata item dalam peringkat yang dipersonalisasi

Jika Anda mengaktifkan metadata dalam rekomendasi untuk kampanye, Anda dapat menentukan kolom metadata kumpulan data Item yang akan disertakan dalam respons. Untuk informasi tentang mengaktifkan metadata, lihat. Metadata item dalam rekomendasi

Contoh kode berikut menunjukkan cara menentukan kolom metadata sebagai bagian dari permintaan Anda untuk peringkat yang dipersonalisasi.

```
aws personalize-runtime get-personalized-ranking \
--campaign-arn Campaign ARN \
--user-id 12 \
--input-list 3 4 10 8 12 7
--metadata-columns "{\"ITEMS\": [\"\"columnNameA\"\",\"\"columnNameB\"\"]}"
```

Mendapatkan peringkat yang dipersonalisasi (AWS SDKs)

Contoh kode berikut menunjukkan bagaimana variasi yang berbeda tentang cara mendapatkan peringkat yang dipersonalisasi dengan AWS SDKs.

Topik

- Mendapatkan peringkat yang dipersonalisasi
- Termasuk metadata item dalam peringkat yang dipersonalisasi
- Mendapatkan peringkat yang dipersonalisasi menggunakan metadata kontekstual

Mendapatkan peringkat yang dipersonalisasi

Kode berikut menunjukkan cara mendapatkan peringkat yang dipersonalisasi untuk pengguna. Tentukan ID pengguna dan daftar item IDs yang akan diberi peringkat untuk pengguna. Item IDs harus dalam data yang Anda gunakan untuk melatih versi solusi. Daftar rekomendasi peringkat dikembalikan. Amazon Personalize mempertimbangkan item pertama dalam daftar yang paling menarik bagi pengguna.

SDK for Python (Boto3)

```
import boto3

personalizeRt = boto3.client('personalize-runtime')

response = personalizeRt.get_personalized_ranking(
    campaignArn = "Campaign arn",
    userId = "UserID",
    inputList = ['ItemID1','ItemID2']
)

print("Personalized Ranking")
```

```
for item in response['personalizedRanking']:
    print (item['itemId'])
```

SDK for Java 2.x

```
public static List<PredictedItem> getRankedRecs(PersonalizeRuntimeClient
 personalizeRuntimeClient,
                                                 String campaignArn,
                                                 String userId,
                                                 ArrayList<String> items) {

    try {
        GetPersonalizedRankingRequest rankingRecommendationsRequest =
 GetPersonalizedRankingRequest.builder()
                .campaignArn(campaignArn)
                .userId(userId)
                .inputList(items)
                .build();

        GetPersonalizedRankingResponse recommendationsResponse =

 personalizeRuntimeClient.getPersonalizedRanking(rankingRecommendationsRequest);
        List<PredictedItem> rankedItems =
 recommendationsResponse.personalizedRanking();
        int rank = 1;
        for (PredictedItem item : rankedItems) {
            System.out.println("Item ranked at position " + rank + " details");
            System.out.println("Item Id is : " + item.itemId());
            System.out.println("Item score is : " + item.score());
            System.out.println("-------------------------------------------");
            rank++;
        }
        return rankedItems;
    } catch (PersonalizeRuntimeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return null;
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
```

```
import { GetPersonalizedRankingCommand } from "@aws-sdk/client-personalize-runtime";
import { personalizeRuntimeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region:
 "REGION"});

// Set the ranking request parameters.
export const getPersonalizedRankingParam = {
  campaignArn: "CAMPAIGN_ARN" /* required */,
  userId: "USER_ID" /* required */,
  inputList: ["ITEM_ID_1", "ITEM_ID_2", "ITEM_ID_3", "ITEM_ID_4"],
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(
      new GetPersonalizedRankingCommand(getPersonalizedRankingParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Termasuk metadata item dalam peringkat yang dipersonalisasi

Jika Anda mengaktifkan metadata dalam rekomendasi untuk kampanye, Anda dapat menentukan kolom metadata kumpulan data Item yang akan disertakan dalam respons. Untuk informasi tentang mengaktifkan metadata, lihat. Metadata item dalam rekomendasi

Contoh kode berikut menunjukkan cara menentukan kolom metadata sebagai bagian dari permintaan Anda untuk peringkat yang dipersonalisasi.

```
import boto3

personalizeRt = boto3.client('personalize-runtime')

response = personalizeRt.get_personalized_ranking(
    campaignArn = "Campaign arn",
```

```
    userId = "UserID",
    inputList = ['ItemID1','ItemID2'],
    metadataColumns = {
      "ITEMS": ['columnNameA','columnNameB']
    }
)

print("Personalized Ranking")
for item in response['personalizedRanking']:
    print (item['itemId'])
    print (item['metadata'])
```

Mendapatkan peringkat yang dipersonalisasi menggunakan metadata kontekstual

Gunakan kode berikut untuk mendapatkan peringkat yang dipersonalisasi berdasarkan metadata kontekstual. Untuk`context`, untuk setiap pasangan kunci-nilai, berikan bidang metadata sebagai kunci dan data konteks sebagai nilai. Dalam kode contoh berikut, kuncinya adalah `DEVICE` dan nilainya adalah`mobile phone`. Ganti nilai-nilai ini dan `Campaign ARN` dan `User ID` dengan nilai-nilai Anda sendiri. Juga ubah `inputList` ke daftar item IDs yang ada di data yang Anda gunakan untuk melatih solusi. Amazon Personalize mempertimbangkan item pertama dalam daftar yang paling menarik bagi pengguna.

```
import boto3

personalizeRt = boto3.client('personalize-runtime')

response = personalizeRt.get_personalized_ranking(
    campaignArn = "Campaign ARN",
    userId = "User ID",
    inputList = ['ItemID1', 'ItemID2'],
    context = {
      'DEVICE': 'mobile phone'
    }
)

print("Personalized Ranking")
for item in response['personalizedRanking']:
  print(item['itemId'])
```

Notebook sampel Peringkat yang Dipersonalisasi

Untuk contoh buku catatan Jupyter yang menunjukkan cara menggunakan resep Personalized-Ranking, lihat Personalize Ranking Example.

Meningkatkan relevansi rekomendasi dengan metadata kontekstual

Untuk meningkatkan relevansi rekomendasi, sertakan metadata kontekstual untuk pengguna, seperti jenis perangkat mereka atau waktu hari, ketika Anda mendapatkan rekomendasi item atau mendapatkan peringkat yang dipersonalisasi.

Untuk menggunakan metadata kontekstual, skema kumpulan data interaksi Item harus memiliki bidang metadata untuk data kontekstual. Misalnya, bidang DEVICE (lihatMembuat file JSON skema untuk skema Amazon Personalize).

Untuk grup kumpulan data Domain, kasus penggunaan pemberi rekomendasi berikut dapat menggunakan metadata kontekstual:

- Direkomendasikan untuk Anda(Domain eCommerce)
- Pilihan teratas untuk Anda(Domain VIDEO_ON_DEMAND)

Untuk sumber daya khusus, resep yang menggunakan metadata kontekstual mencakup yang berikut:

- Personalisasi pengguna-v2 dan Personalisasi Pengguna
- Personalisasi-peringkat-v2 dan Peringkat yang Dipersonalisasi

Untuk informasi selengkapnya tentang informasi kontekstual, lihat postingan Blog AWS Machine Learning berikut: Meningkatkan relevansi rekomendasi Amazon Personalize Anda dengan memanfaatkan informasi kontekstual.

Anda bisa mendapatkan rekomendasi dengan metadata kontekstual dengan konsol Amazon Personalize, (), atau. AWS Command Line Interface AWS CLI AWS SDKs

Mendapatkan rekomendasi menggunakan metadata kontekstual (Python SDK)AWS

Untuk meningkatkan relevansi rekomendasi, sertakan metadata kontekstual untuk pengguna, seperti jenis perangkat mereka atau waktu hari, ketika Anda mendapatkan rekomendasi item atau mendapatkan peringkat yang dipersonalisasi.

Gunakan kode berikut untuk mendapatkan rekomendasi berdasarkan metadata kontekstual. Untuk`context`, untuk setiap pasangan kunci-nilai, berikan bidang metadata sebagai kunci dan data konteks sebagai nilai. Dalam kode contoh berikut, kuncinya adalah `DEVICE` dan nilainya adalah`mobile phone`. Ganti nilai-nilai ini dan `Campaign ARN` dan `User ID` dengan nilai-nilai Anda sendiri. Jika Anda membuat pemberi rekomendasi, ganti `campaignArn` dengan`recommenderArn`. Daftar item yang direkomendasikan untuk ditampilkan pengguna.

```
import boto3

personalizeRt = boto3.client('personalize-runtime')

response = personalizeRt.get_recommendations(
    campaignArn = 'Campaign ARN',
    userId = 'User ID',
    context = {
      'DEVICE': 'mobile phone'
    }
)

print("Recommended items")
for item in response['itemList']:
    print (item['itemId'])
```

Mendapatkan rekomendasi item batch dengan sumber daya khusus

Dengan sumber daya khusus, Anda bisa mendapatkan rekomendasi item dengan aliran batch asinkron. Misalnya, Anda mungkin mendapatkan rekomendasi produk untuk semua pengguna di daftar email atau item-to-itemkesamaan di seluruh inventaris.

Untuk mendapatkan rekomendasi batch untuk item, Anda menggunakan pekerjaan inferensi batch. Pekerjaan inferensi batch adalah alat yang mengimpor data input batch Anda dari bucket Amazon

S3, menggunakan versi solusi kustom Anda untuk menghasilkan rekomendasi item, lalu mengekspor rekomendasi item ke bucket Amazon S3. Bergantung pada resepnya, data masukan Anda adalah daftar pengguna, atau item, atau daftar pengguna masing-masing dengan kumpulan item.

Jika solusi Anda menggunakan resep Item Serupa dan Anda memiliki kumpulan data Item dengan data tekstual dan data judul item, Anda dapat membuat rekomendasi batch dengan tema untuk setiap grup item. Untuk informasi selengkapnya, lihat Rekomendasi Batch dengan tema dari Content Generator.

Setelah Anda membuat versi solusi kustom, bagaimana data baru memengaruhi rekomendasi item batch bergantung pada jenisnya, metode impor, dan resep khusus yang Anda gunakan. Untuk informasi tentang bagaimana data baru memengaruhi rekomendasi batch, lihatBagaimana data baru memengaruhi rekomendasi batch (sumber daya khusus).

Topik

- Alur kerja Batch
- Pedoman dan persyaratan
- Penilaian alur kerja Batch
- Rekomendasi Batch dengan tema dari Content Generator
- Mempersiapkan data input untuk rekomendasi batch
- Membuat pekerjaan inferensi batch
- Contoh keluaran pekerjaan inferensi Batch

Alur kerja Batch

Alur kerja batch adalah sebagai berikut:

1. Siapkan dan unggah data masukan Anda dalam format JSON ke bucket Amazon S3. Format data input Anda tergantung pada resep yang Anda gunakan. Lihat Mempersiapkan data input untuk rekomendasi batch.
2. Buat lokasi terpisah untuk data keluaran Anda, baik folder atau bucket Amazon S3 lainnya.
3. Buat pekerjaan inferensi batch. Lihat Membuat pekerjaan inferensi batch.
4. Saat inferensi batch selesai, ambil rekomendasi item dari lokasi keluaran Anda di Amazon S3.

Pedoman dan persyaratan

Berikut ini adalah pedoman dan persyaratan untuk mendapatkan rekomendasi batch:

- Peran layanan Amazon Personalisasi IAM Anda harus memiliki izin untuk membaca dan menambahkan file ke bucket Amazon S3 Anda. Untuk informasi tentang pemberian izin, lihat. Kebijakan peran layanan untuk alur kerja batch Untuk informasi selengkapnya tentang izin bucket, lihat contoh kebijakan pengguna di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon. Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.
- Anda harus membuat solusi kustom dan versi solusi sebelum Anda membuat pekerjaan inferensi batch. Namun, Anda tidak perlu membuat kampanye Amazon Personalize. Jika Anda membuat grup kumpulan data Domain, Anda masih dapat membuat sumber daya kustom.
- Untuk menghasilkan tema dengan rekomendasi, Anda harus menggunakan resep Similar-Items. Dan Anda harus memiliki kumpulan data Item dengan data tekstual dan data judul item. Untuk informasi lebih lanjut tentang rekomendasi bertema, lihatRekomendasi Batch dengan tema dari Content Generator.
- Data masukan Anda harus diformat seperti yang dijelaskan dalamMempersiapkan data input untuk segmen pengguna.
- Anda tidak bisa mendapatkan rekomendasi batch dengan Trending-Now atau resep. Next-Best-Action
- Jika Anda menggunakan filter dengan parameter placeholder, Anda harus menyertakan nilai untuk parameter dalam data input Anda dalam objek. `filterValues` Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda.
- Kami menyarankan Anda menggunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda) daripada data input Anda.
- Rekomendasi Batch mungkin tidak persis sama dengan rekomendasi waktu nyata. Ini karena pekerjaan inferensi batch membutuhkan waktu lebih lama untuk diselesaikan dan hanya mempertimbangkan data yang tersedia 15 menit sebelum dimulainya pekerjaan.

Penilaian alur kerja Batch

Rekomendasi Batch mencakup skor sebagai berikut:

- Dengan resep Personalisasi Pengguna dan Personalisasi Peringkat, Amazon Personalize menghitung skor rekomendasi pekerjaan inferensi batch seperti yang dijelaskan dalam dan. Cara kerja penilaian rekomendasi (sumber daya khusus) Cara kerja penilaian peringkat yang dipersonalisasi Anda dapat melihat skor dalam file JSON keluaran pekerjaan inferensi batch.
- Dengan resep Similar-Items, jika Anda mendapatkan rekomendasi batch bertema, Amazon Personalize memberi peringkat setiap set item terkait berdasarkan seberapa relevan tema untuk setiap item. Setiap item termasuk skor dari 0 hingga 1. Semakin tinggi skor, semakin erat hubungannya item dengan tema. Untuk informasi selengkapnya tentang rekomendasi dengan tema, lihatRekomendasi Batch dengan tema dari Content Generator.

Rekomendasi Batch dengan tema dari Content Generator

> **Important**
>
> Ketika Anda mendapatkan rekomendasi batch dengan tema, Anda dikenakan biaya tambahan. Untuk informasi selengkapnya, lihat Harga Amazon Personalisasi.

Jika Anda menggunakanResep Barang Serupa, Amazon Personalize Content Generator dapat menambahkan tema deskriptif ke rekomendasi batch. Content Generator adalah kemampuan kecerdasan buatan generatif (AI generatif) yang dikelola oleh Amazon Personalize.

Saat Anda mendapatkan rekomendasi batch dengan tema, Amazon Personalize Content Generator menambahkan tema deskriptif untuk setiap set item serupa. Tema ini didasarkan pada deskripsi item dan data nama item dalam dataset Item Anda. Amazon Personalize menyertakan tema dalam output pekerjaan inferensi batch. Anda dapat menggunakan tema untuk membuat teks dalam aplikasi atau pesan pemasaran Anda lebih menarik.

Misalnya, jika Anda mendapatkan rekomendasi item terkait untuk item makanan sarapan, Amazon Personalize mungkin menghasilkan tema seperti Rise and shine atau Morning Essentials. Anda dapat menggunakan tema untuk mengganti judul korsel generik, seperti Sering dibeli bersama. Atau Anda dapat memasukkan tema dalam email promosi atau kampanye pemasaran untuk opsi menu baru.

AWS tidak memantau tema dari Content Generator. Untuk mengkonfirmasi kualitas tema, Anda dapat menggunakan skor yang dihasilkan untuk setiap item yang direkomendasikan. Untuk informasi selengkapnya, lihat Peringkat dan penilaian untuk rekomendasi batch dengan tema.

Topik

- Wilayah yang didukung
- Pedoman dan persyaratan
- Peringkat dan penilaian untuk rekomendasi batch dengan tema
- Menghasilkan rekomendasi batch dengan tema

Wilayah yang didukung

Amazon Personalize Content Generator hanya tersedia di Wilayah berikut: AWS

- AS Timur (Virginia Utara)
- AS Barat (Oregon)
- Asia Pasifik (Tokyo)

Pedoman dan persyaratan

Berikut ini adalah pedoman dan persyaratan untuk menghasilkan rekomendasi dengan tema:

- File input Anda dapat memiliki hingga 100 item. Untuk informasi tentang data masukan untuk rekomendasi batch, lihatMempersiapkan data input untuk rekomendasi batch.
- Solusi Anda harus menggunakanResep Barang Serupa.
- Anda harus memiliki kumpulan data Item dengan data berikut. Data ini dapat membantu menghasilkan tema yang lebih relevan.
  - Itu harus memiliki bidang tekstual, seperti bidang DESKRIPSI. Untuk informasi tentang data tekstual, lihatMetadata teks tidak terstruktur.
  - Itu harus memiliki kolom string dengan data nama item, seperti bidang TITLE.

  Jika kumpulan data Item Anda tidak memiliki data ini, Anda dapat menambahkannya. Untuk informasi tentang memperbarui data yang ada, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Peringkat dan penilaian untuk rekomendasi batch dengan tema

Saat Anda mendapatkan rekomendasi batch dengan tema, Amazon Personalize memberi peringkat setiap set item berdasarkan seberapa relevan tema untuk setiap item. Setiap item mencakup skor dalam kisaran kasar -0,1 dan 0,6. Semakin tinggi skor, semakin erat hubungannya item dengan tema.

Anda dapat menggunakan skor untuk menetapkan ambang batas untuk hanya menampilkan item yang sangat terkait dengan tema.

Misalnya, Amazon Personalize mungkin mengembalikan tema`For your sweet tooth`, dan item terkait serta skornya mungkin: permen keras (skor 0,19884521), cokelat (skor .17664525), apel (skor .08994528), es loli (skor .14294521), ubi jalar (skor .07794527), dan wortel (skor .04994523). Dalam aplikasi Anda, Anda dapat menambahkan aturan untuk memasukkan hanya item dengan skor `.10` atau lebih besar, menghilangkan buah dan sayuran.

Contoh berikut menunjukkan format output pekerjaan inferensi batch yang menghasilkan rekomendasi film dengan tema.

```
{"input":{"itemId":"40"},"output":{"recommendedItems":
["36","50","44","22","21","29","3","1","2","39"],"theme":"Movies
 with a strong female lead","itemsThemeRelevanceScores":
[0.19994527,0.183059963,0.17478035,0.1618133,0.1574806,0.15468733,0.1499242,0.14353688,0.135314
{"input":{"itemId":"43"},"output":{"recommendedItems":
["50","21","36","3","17","2","39","1","10","5"],"theme":"The best movies of
 1995","itemsThemeRelevanceScores":
[0.184988,0.1795761,0.11143453,0.0989443,0.08258403,0.07952615,0.07115086,0.0621634,-0.138913,-
...
```

Menghasilkan rekomendasi batch dengan tema

Untuk menghasilkan rekomendasi batch dengan tema, Anda menyelesaikan alur kerja batch seperti yang dijelaskan dalamAlur kerja Batch. Anda menyiapkan data input Anda dengan cara yang sama seperti yang Anda lakukan untuk `RELATED_ITEMS` resep. Sebagai contoh, lihat Resep RELATED_ITEMS.

Saat Anda membuat pekerjaan inferensi batch, Anda mengaktifkan pembuatan tema dan menentukan kolom judul item dari kumpulan data Item Anda.

- Untuk informasi tentang menggunakan konsol Amazon Personalize untuk membuat pekerjaan inferensi batch yang menghasilkan tema, lihat. Membuat pekerjaan inferensi batch
- Untuk contoh kode yang menunjukkan cara menggunakan SDK for Python (Boto3) untuk membuat pekerjaan inferensi batch yang menghasilkan tema, lihat. Membuat pekerjaan inferensi batch yang menghasilkan tema

Mempersiapkan data input untuk rekomendasi batch

Pekerjaan inferensi batch mengimpor data JSON masukan batch Anda dari bucket Amazon S3, menggunakan versi solusi kustom Anda untuk menghasilkan rekomendasi, lalu mengekspor rekomendasi item ke bucket Amazon S3. Sebelum Anda bisa mendapatkan rekomendasi batch, Anda harus menyiapkan dan mengunggah file JSON Anda ke bucket Amazon S3. Sebaiknya buat folder keluaran di bucket Amazon S3 atau gunakan bucket Amazon S3 keluaran terpisah. Anda kemudian dapat menjalankan beberapa pekerjaan inferensi batch menggunakan lokasi data input yang sama.

Jika Anda menggunakan filter dengan parameter placeholder, seperti`$GENRE`, Anda harus memberikan nilai untuk parameter dalam `filterValues` objek di JSON input Anda. Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda.

Untuk menyiapkan dan mengimpor data

1. Format data input batch Anda tergantung pada resep Anda. Anda tidak bisa mendapatkan rekomendasi batch dengan resep Trending-Now.

   - Untuk resep USER_PERSONALIZATION dan resep Popularity-Count, data input Anda adalah file JSON dengan daftar UserIds
   - Untuk resep RELATED_ITEMS, data masukan Anda adalah daftar ItemIds
   - Untuk resep PERSONALIZED_RANKING, data masukan Anda adalah daftar UserIds, masing-masing dipasangkan dengan koleksi ItemIds

   Pisahkan setiap baris dengan baris baru. Untuk contoh data masukan, lihatContoh input dan keluaran pekerjaan inferensi Batch JSON.
2. Unggah JSON masukan Anda ke folder input di bucket Amazon S3 Anda. Untuk informasi selengkapnya, lihat Mengunggah file dan folder menggunakan seret dan lepas di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon
3. Buat lokasi terpisah untuk data keluaran Anda, baik folder atau bucket Amazon S3 lainnya. Dengan membuat lokasi terpisah untuk JSON keluaran, Anda dapat menjalankan beberapa pekerjaan inferensi batch dengan lokasi data input yang sama.
4. Buat pekerjaan inferensi batch. Amazon Personalize mengeluarkan rekomendasi dari versi solusi Anda ke lokasi data keluaran Anda.

Contoh input dan keluaran pekerjaan inferensi Batch JSON

Bagaimana Anda memformat data input Anda resep yang Anda gunakan. Jika Anda menggunakan filter dengan parameter placeholder, seperti`$GENRE`, Anda harus memberikan nilai untuk parameter dalam `filterValues` objek di JSON input Anda. Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda.

Bagian berikut mencantumkan contoh input dan output JSON yang diformat dengan benar untuk pekerjaan inferensi batch. Anda tidak bisa mendapatkan rekomendasi batch dengan resep Trending-Now.

Topik

- Resep USER_PERSONALIZATION
- Resep POPULAR_ITEMS (Hanya Hitungan Popularitas)
- resep PERSONALIZED_RANKING
- Resep RELATED_ITEMS

Resep USER_PERSONALIZATION

Berikut ini menunjukkan contoh input dan output JSON yang diformat dengan benar untuk resep USER_PERSONALIZATION. Jika Anda menggunakan User-Personalization-v 2, setiap item yang direkomendasikan mencakup daftar alasan mengapa item tersebut dimasukkan dalam rekomendasi. Daftar ini bisa kosong. Untuk informasi tentang kemungkinan alasan, lihatAlasan rekomendasi dengan User-Personalization-v 2.

Input

Pisahkan masing-masing `userId` dengan baris baru sebagai berikut.

```
{"userId": "4638"}
{"userId": "663"}
{"userId": "3384"}
...
```

Output

```
{"input":{"userId":"4638"},"output":{"recommendedItems":
["63992","115149","110102","148626","148888","31685","102445","69526","92535","143355","6237
[0.0152238,0.0069081,0.0068222,0.006394,0.0059746,0.0055851,0.0049357,0.0044644,0.0042968,0.
```

```
{"input":{"userId":"663"},"output":{"recommendedItems":
["368","377","25","780","1610","648","1270","6","165","1196","1097","300","1183","608","104"
[0.0406197,0.0372557,0.0254077,0.0151975,0.014991,0.0127175,0.0124547,0.0116712,0.0091098,0.
{"input":{"userId":"3384"},"output":{"recommendedItems":
["597","21","223","2144","208","2424","594","595","920","104","520","367","2081","39","1035"
[0.0241061,0.0119394,0.0118012,0.010662,0.0086972,0.0079428,0.0073218,0.0071438,0.0069602,0.
...
```

Resep POPULAR_ITEMS (Hanya Hitungan Popularitas)

Berikut ini menunjukkan contoh input dan output JSON yang diformat dengan benar untuk resep Popularity-Count. Anda tidak bisa mendapatkan rekomendasi batch dengan resep Trending-Now.

Input

Pisahkan masing-masing `userId` dengan baris baru sebagai berikut.

```
{"userId": "12"}
{"userId": "105"}
{"userId": "41"}
...
```

Output

```
{"input": {"userId": "12"}, "output": {"recommendedItems": ["105", "106", "441"]}}
{"input": {"userId": "105"}, "output": {"recommendedItems": ["105", "106", "441"]}}
{"input": {"userId": "41"}, "output": {"recommendedItems": ["105", "106", "441"]}}
...
```

resep PERSONALIZED_RANKING

Berikut ini menunjukkan contoh input dan output JSON yang diformat dengan benar untuk resep PERSONALIZED_RANKING.

Input

Pisahkan masing-masing `userId` dan daftar `itemIds` yang akan diberi peringkat dengan baris baru sebagai berikut.

```
{"userId": "891", "itemList": ["27", "886", "101"]}
{"userId": "445", "itemList": ["527", "55", "901"]}
```

```
{"userId": "71", "itemList": ["27", "351", "101"]}
...
```

Output

```
{"input":{"userId":"891","itemList":["27","886","101"]},"output":
{"recommendedItems":["27","101","886"],"scores":[0.48421,0.28133,0.23446]}}
{"input":{"userId":"445","itemList":["527","55","901"]},"output":
{"recommendedItems":["901","527","55"],"scores":[0.46972,0.31011,0.22017]}}
{"input":{"userId":"71","itemList":["29","351","199"]},"output":{"recommendedItems":
["351","29","199"],"scores":[0.68937,0.24829,0.06232]}}
...
```

Resep RELATED_ITEMS

Berikut ini menunjukkan contoh input dan output JSON yang diformat dengan benar untuk resep RELATED_ITEMS.

Input

Pisahkan masing-masing `itemId` dengan baris baru sebagai berikut.

```
{"itemId": "105"}
{"itemId": "106"}
{"itemId": "441"}
...
```

Output

```
{"input": {"itemId": "105"}, "output": {"recommendedItems": ["106", "107", "49"]}}
{"input": {"itemId": "106"}, "output": {"recommendedItems": ["105", "107", "49"]}}
{"input": {"itemId": "441"}, "output": {"recommendedItems": ["2", "442", "435"]}}
...
```

Berikut ini menunjukkan contoh input dan output JSON yang diformat dengan benar untuk resep Similar-Items dengan tema.

Input

Pisahkan masing-masing `itemId` dengan baris baru sebagai berikut.

```
{"itemId": "40"}
{"itemId": "43"}
...
```

Output

```
{"input":{"itemId":"40"},"output":{"recommendedItems":
["36","50","44","22","21","29","3","1","2","39"],"theme":"Movies
 with a strong female lead","itemsThemeRelevanceScores":
[0.19994527,0.183059963,0.17478035,0.1618133,0.1574806,0.15468733,0.1499242,0.14353688,0.135
{"input":{"itemId":"43"},"output":{"recommendedItems":
["50","21","36","3","17","2","39","1","10","5"],"theme":"The best movies of
 1995","itemsThemeRelevanceScores":
[0.184988,0.1795761,0.11143453,0.0989443,0.08258403,0.07952615,0.07115086,0.0621634,-0.13891
...
```

Membuat pekerjaan inferensi batch

Buat tugas inferensi batch untuk mendapatkan rekomendasi item batch bagi pengguna berdasarkan data masukan dari Amazon S3. Data input dapat berupa daftar pengguna atau item (atau keduanya) dalam format JSON. Anda dapat membuat pekerjaan inferensi batch dengan konsol Amazon Personalize, AWS Command Line Interface the AWS CLI(), atau. AWS SDKs

Saat membuat pekerjaan inferensi batch, Anda menentukan jalur Amazon S3 ke lokasi input dan output. Amazon S3 berbasis awalan. Jika Anda memberikan awalan untuk lokasi data input, Amazon Personalize menggunakan semua file yang cocok dengan awalan tersebut sebagai data input. Misalnya, jika Anda menyediakan `s3://amzn-s3-demo-bucket/folderName` dan bucket Anda juga memiliki folder dengan jalurs`3://amzn-s3-demo-bucket/folderName_test`, Amazon Personalize menggunakan semua file di kedua folder sebagai data input. Untuk hanya menggunakan file dalam folder tertentu sebagai data input, akhiri jalur Amazon S3 dengan pembatas awalan, seperti/: `s3://amzn-s3-demo-bucket/folderName/` Untuk informasi selengkapnya tentang cara Amazon S3 mengatur objek, lihat Mengatur, mencantumkan, dan bekerja dengan objek Anda.

Untuk informasi selengkapnya tentang alur kerja batch di Amazon Personalize, termasuk persyaratan izin, penilaian rekomendasi, serta menyiapkan serta mengimpor data input, lihat. Mendapatkan rekomendasi item batch dengan sumber daya khusus

Topik

- Membuat pekerjaan inferensi batch (konsol)

- Membuat pekerjaan inferensi batch ()AWS CLI
- Membuat pekerjaan inferensi batch ()AWS SDKs

Membuat pekerjaan inferensi batch (konsol)

Setelah selesaiMempersiapkan data input untuk rekomendasi batch, Anda siap untuk membuat pekerjaan inferensi batch. Prosedur ini mengasumsikan bahwa Anda telah membuat solusi dan versi solusi (model terlatih).

Untuk membuat pekerjaan inferensi batch (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
3. Dari panel navigasi, di bawah Sumber daya khusus, pilih pekerjaan inferensi Batch.
4. Pilih Buat pekerjaan inferensi batch.
5. Pilih jenis pekerjaan inferensi batch.
   - Untuk menghasilkan rekomendasi item tanpa tema, pilih Rekomendasi item.
   - Jika Anda menggunakan resep Similar-Items dan ingin menambahkan tema deskriptif ke grup item serupa, pilih Rekomendasi bertema dengan Content Generator. Untuk menghasilkan tema, Anda harus memiliki kumpulan data Item dengan data nama item dan data tekstual. Untuk informasi selengkapnya, lihat Rekomendasi Batch dengan tema dari Content Generator.
6. Dalam detail pekerjaan inferensi Batch, dalam nama pekerjaan inferensi Batch, tentukan nama untuk pekerjaan inferensi batch Anda.
7. Untuk Solusi, pilih solusi dan kemudian pilih ID versi Solusi yang ingin Anda gunakan untuk menghasilkan rekomendasi.
8. Untuk Jumlah hasil, secara opsional tentukan jumlah rekomendasi untuk setiap baris data input. Default-nya adalah 25.
9. Jika pekerjaan batch Anda menghasilkan rekomendasi dengan tema, di Detail rekomendasi bertema, pilih kolom yang berisi nama atau judul untuk item dalam kumpulan data Item Anda. Data ini dapat membantu menghasilkan tema yang lebih relevan. Untuk informasi selengkapnya, lihat Rekomendasi Batch dengan tema dari Content Generator.
10. Di Sumber input, tentukan jalur Amazon S3 ke file input Anda.

Gunakan sintaks berikut: **`s3://amzn-s3-demo-bucket/<folder name>/<input JSON file name>.json`**

Data masukan Anda harus dalam format yang benar untuk resep yang digunakan solusi Anda. Untuk contoh data masukan lihatContoh input dan keluaran pekerjaan inferensi Batch JSON.

11. Untuk kunci Dekripsi, jika Anda menggunakan AWS KMS kunci Anda sendiri untuk enkripsi bucket, tentukan Nama Sumber Daya Amazon (ARN) kunci Anda. Amazon Personalize harus memiliki izin untuk menggunakan kunci Anda. Untuk informasi tentang pemberian izin, lihat. Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS
12. Di Tujuan keluaran, tentukan jalur ke lokasi keluaran Anda. Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda).

    Gunakan sintaks berikut: **`s3://amzn-s3-demo-bucket/<output folder name>/`**
13. Untuk kunci Enkripsi, jika Anda menggunakan AWS KMS kunci Anda sendiri untuk enkripsi, tentukan ARN kunci Anda. Amazon Personalize harus memiliki izin untuk menggunakan kunci Anda. Untuk informasi tentang pemberian izin, lihat. Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS
14. Untuk peran layanan IAM, pilih peran layanan IAM yang Anda buat untuk Amazon Personalisasi selama penyiapan. Peran ini harus memiliki akses baca dan tulis ke bucket Amazon S3 masukan dan keluaran Anda masing-masing.
15. Di Filter opsional pilih filter untuk menerapkan filter ke rekomendasi batch. Jika filter Anda menggunakan parameter placeholder, pastikan nilai untuk parameter disertakan dalam input JSON Anda. Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda.
16. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya
17. Pilih Buat pekerjaan inferensi batch. Pembuatan pekerjaan inferensi Batch dimulai dan halaman pekerjaan inferensi Batch muncul dengan bagian detail pekerjaan inferensi Batch ditampilkan.

    Saat status pekerjaan inferensi batch berubah menjadi Aktif, Anda dapat mengambil output pekerjaan dari bucket Amazon S3 keluaran yang ditentukan. Nama file output akan menjadi format*`input-name`*`.out`.

Membuat pekerjaan inferensi batch ()AWS CLI

Setelah selesaiMempersiapkan data input untuk rekomendasi batch, Anda siap untuk membuat pekerjaan inferensi batch dengan CreateBatchInferenceJob operasi.

Topik

- Membuat pekerjaan inferensi batch
- Membuat pekerjaan inferensi batch yang menghasilkan tema

Membuat pekerjaan inferensi batch

Anda dapat menggunakan `create-batch-inference-job` perintah untuk membuat pekerjaan inferensi batch. Tentukan nama pekerjaan, ganti `Solution version ARN` dengan Amazon Resource Name (ARN) versi solusi Anda, dan ganti `IAM service role ARN` dengan ARN peran layanan IAM yang Anda buat untuk Amazon Personalize selama penyiapan. Peran ini harus memiliki akses baca dan tulis ke bucket Amazon S3 masukan dan keluaran Anda masing-masing. Secara opsional menyediakan filter ARN untuk memfilter rekomendasi. Jika filter Anda menggunakan parameter placeholder, pastikan nilai untuk parameter disertakan dalam input JSON Anda. Untuk informasi selengkapnya, lihat Memfilter rekomendasi batch dan segmen pengguna (sumber daya khusus).

Ganti `S3 input path` dan `S3 output path` dengan jalur Amazon S3 ke file input dan lokasi output Anda. Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda). Gunakan sintaks berikut untuk lokasi input dan output: **`s3://amzn-s3-demo-bucket/<folder name>/<input JSON file name>.json`** dan**`s3://amzn-s3-demo-bucket/<output folder name>/`**.

Contohnya termasuk `itemExplorationConfig` hiperparameter khusus resep Personalisasi Pengguna opsional: dan. `explorationWeight` `explorationItemAgeCutOff` Secara opsional sertakan `explorationWeight` dan `explorationItemAgeCutOff` nilai untuk mengonfigurasi eksplorasi. Untuk informasi selengkapnya, lihat Resep Personalisasi Pengguna.

```
aws personalize create-batch-inference-job \
--job-name Batch job name \
--solution-version-arn Solution version ARN \
--filter-arn Filter ARN \
--job-input s3DataSource={path=s3://S3 input path} \
--job-output s3DataDestination={path=s3://S3 output path} \
--role-arn IAM service role ARN \
```

```
--batch-inference-job-config "{\"itemExplorationConfig\":{\"explorationWeight\":
\"0.3\",\"explorationItemAgeCutOff\":\"30\"}}"
```

Membuat pekerjaan inferensi batch yang menghasilkan tema

Untuk menghasilkan tema untuk item serupa, Anda harus menggunakan resep Item Serupa dan kumpulan data Item Anda harus memiliki bidang tekstual dan kolom data nama item. Untuk informasi selengkapnya tentang rekomendasi dengan tema, lihatRekomendasi Batch dengan tema dari Content Generator.

Kode berikut membuat pekerjaan inferensi batch yang menghasilkan rekomendasi dengan tema. Biarkan `batch-inference-job-mode` set ke`THEME_GENERATION`. Ganti `COLUMN_NAME` dengan nama kolom yang menyimpan data nama item Anda.

```
aws personalize create-batch-inference-job \
--job-name Themed batch job name \
--solution-version-arn Solution version ARN \
--filter-arn Filter ARN \
--job-input s3DataSource={path=s3://S3 input path} \
--job-output s3DataDestination={path=s3://S3 output path} \
--role-arn IAM service role ARN \
--batch-inference-job-mode THEME_GENERATION \
--theme-generation-config "{\"fieldsForThemeGeneration\": {\"itemName\":
\"COLUMN_NAME\"}}"
```

Membuat pekerjaan inferensi batch ()AWS SDKs

Setelah selesaiMempersiapkan data input untuk rekomendasi batch, Anda siap untuk membuat pekerjaan inferensi batch dengan CreateBatchInferenceJob operasi.

Topik

- Membuat pekerjaan inferensi batch
- Membuat pekerjaan inferensi batch yang menghasilkan tema

Membuat pekerjaan inferensi batch

Anda dapat menggunakan kode berikut untuk membuat pekerjaan inferensi batch. Tentukan nama pekerjaan, Nama Sumber Daya Amazon (ARN) versi solusi Anda, dan ARN peran layanan IAM yang Anda buat untuk Amazon Personalisasi selama penyiapan. Peran ini harus memiliki akses baca dan tulis ke bucket Amazon S3 input dan output Anda.

Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda). Gunakan sintaks berikut untuk lokasi input dan output: **`s3:/amzn-s3-demo-bucket/<folder name>/<input JSON file name>.json`** dan**`s3://amzn-s3-demo-bucket/<output folder name>/`**.

Untuk`numResults`, tentukan jumlah item yang ingin diprediksi Amazon Personalize untuk setiap baris data input. Secara opsional menyediakan filter ARN untuk memfilter rekomendasi. Jika filter Anda menggunakan parameter placeholder, pastikan nilai untuk parameter disertakan dalam input JSON Anda. Untuk informasi selengkapnya, lihat Memfilter rekomendasi batch dan segmen pengguna (sumber daya khusus).

SDK for Python (Boto3)

Contohnya termasuk `itemExplorationConfig` hiperparameter khusus resep Personalisasi Pengguna opsional: dan. `explorationWeight` `explorationItemAgeCutOff` Secara opsional sertakan `explorationWeight` dan `explorationItemAgeCutOff` nilai untuk mengonfigurasi eksplorasi. Untuk informasi selengkapnya, lihat Resep Personalisasi Pengguna.

```
import boto3

personalize_rec = boto3.client(service_name='personalize')

personalize_rec.create_batch_inference_job (
    solutionVersionArn = "Solution version ARN",
    jobName = "Batch job name",
    roleArn = "IAM service role ARN",
    filterArn = "Filter ARN",
    batchInferenceJobConfig = {
        # optional USER_PERSONALIZATION recipe hyperparameters
        "itemExplorationConfig": {
            "explorationWeight": "0.3",
            "explorationItemAgeCutOff": "30"
        }
    },
    jobInput =
       {"s3DataSource": {"path": "s3://amzn-s3-demo-bucket/<folder name>/<input JSON
 file name>.json"}},
    jobOutput =
       {"s3DataDestination": {"path": "s3:/amzn-s3-demo-bucket/<output folder
 name>/"}}
)
```

SDK for Java 2.x

Contoh ini mencakup `itemExplorationConfig` bidang khusus resep Personalisasi Pengguna opsional: dan. `explorationWeight` `explorationItemAgeCutOff` Secara opsional sertakan `explorationWeight` dan `explorationItemAgeCutOff` nilai untuk mengonfigurasi eksplorasi. Untuk informasi selengkapnya, lihat Resep Personalisasi Pengguna.

```
public static String createPersonalizeBatchInferenceJob(PersonalizeClient
 personalizeClient,
                                                         String solutionVersionArn,
                                                         String jobName,
                                                         String filterArn,
                                                         String
 s3InputDataSourcePath,
                                                         String
 s3DataDestinationPath,
                                                         String roleArn,
                                                         String explorationWeight,
                                                         String
 explorationItemAgeCutOff) {

  long waitInMilliseconds = 60 * 1000;
  String status;
  String batchInferenceJobArn;

  try {
      // Set up data input and output parameters.
      S3DataConfig inputSource = S3DataConfig.builder()
              .path(s3InputDataSourcePath)
              .build();
      S3DataConfig outputDestination = S3DataConfig.builder()
              .path(s3DataDestinationPath)
              .build();

      BatchInferenceJobInput jobInput = BatchInferenceJobInput.builder()
              .s3DataSource(inputSource)
              .build();
      BatchInferenceJobOutput jobOutputLocation = BatchInferenceJobOutput.builder()
              .s3DataDestination(outputDestination)
              .build();

      // Optional code to build the User-Personalization specific item exploration
 config.
```

```
    HashMap<String, String> explorationConfig = new HashMap<>();

    explorationConfig.put("explorationWeight", explorationWeight);
    explorationConfig.put("explorationItemAgeCutOff", explorationItemAgeCutOff);

    BatchInferenceJobConfig jobConfig = BatchInferenceJobConfig.builder()
            .itemExplorationConfig(explorationConfig)
            .build();
    // End optional User-Personalization recipe specific code.

    CreateBatchInferenceJobRequest createBatchInferenceJobRequest =
CreateBatchInferenceJobRequest.builder()
            .solutionVersionArn(solutionVersionArn)
            .jobInput(jobInput)
            .jobOutput(jobOutputLocation)
            .jobName(jobName)
            .filterArn(filterArn)
            .roleArn(roleArn)
            .batchInferenceJobConfig(jobConfig)   // Optional
            .build();

    batchInferenceJobArn =
personalizeClient.createBatchInferenceJob(createBatchInferenceJobRequest)
            .batchInferenceJobArn();
    DescribeBatchInferenceJobRequest describeBatchInferenceJobRequest =
DescribeBatchInferenceJobRequest.builder()
            .batchInferenceJobArn(batchInferenceJobArn)
            .build();

    long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

    // wait until the batch inference job is complete.
    while (Instant.now().getEpochSecond() < maxTime) {

        BatchInferenceJob batchInferenceJob = personalizeClient
                .describeBatchInferenceJob(describeBatchInferenceJobRequest)
                .batchInferenceJob();

        status = batchInferenceJob.status();
        System.out.println("Batch inference job status: " + status);

        if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
            break;
        }
```

```
            try {
                Thread.sleep(waitInMilliseconds);
            } catch (InterruptedException e) {
                System.out.println(e.getMessage());
            }
        }
        return batchInferenceJobArn;

    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateBatchInferenceJobCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the batch inference job's parameters.

export const createBatchInferenceJobParam = {
  jobName: "JOB_NAME",
  jobInput: {
    s3DataSource: {
      path: "INPUT_PATH",
    },
  },
  jobOutput: {
    s3DataDestination: {
      path: "OUTPUT_PATH",
    },
  },
  roleArn: "ROLE_ARN",
  solutionVersionArn: "SOLUTION_VERSION_ARN",
  numResults: 20,
};

export const run = async () => {
```

```
  try {
    const response = await personalizeClient.send(
      new CreateBatchInferenceJobCommand(createBatchInferenceJobParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Memproses pekerjaan batch mungkin membutuhkan waktu beberapa saat untuk diselesaikan. Anda dapat memeriksa status pekerjaan dengan memanggil DescribeBatchInferenceJob dan meneruskan `batchRecommendationsJobArn` sebagai parameter input. Anda juga dapat mencantumkan semua pekerjaan inferensi batch Amazon Personalisasi di AWS lingkungan Anda dengan menelepon. ListBatchInferenceJobs

Membuat pekerjaan inferensi batch yang menghasilkan tema

Untuk menghasilkan tema untuk item serupa, Anda harus menggunakan resep Item Serupa dan kumpulan data Item Anda harus memiliki bidang tekstual dan kolom data nama item. Untuk informasi selengkapnya tentang rekomendasi dengan tema, lihatRekomendasi Batch dengan tema dari Content Generator.

Kode berikut membuat pekerjaan inferensi batch yang menghasilkan rekomendasi dengan tema. Biarkan `batchInferenceJobMode` set ke`"THEME_GENERATION"`. Ganti `COLUMNN_NAME` dengan nama kolom yang menyimpan data nama item Anda.

```
import boto3

personalize_rec = boto3.client(service_name='personalize')

personalize_rec.create_batch_inference_job (
    solutionVersionArn = "Solution version ARN",
    jobName = "Batch job name",
    roleArn = "IAM service role ARN",
    filterArn = "Filter ARN",
    batchInferenceJobMode = "THEME_GENERATION",
    themeGenerationConfig = {
```

```
      "fieldsForThemeGeneration": {
          "itemName": "COLUMN_NAME"
      }
    },
    jobInput =
       {"s3DataSource": {"path": "s3://amzn-s3-demo-bucket/<folder name>/<input JSON
 file name>.json"}},
    jobOutput =
       {"s3DataDestination": {"path": "s3://amzn-s3-demo-bucket/<output folder
 name>/"}}
)
```

Contoh keluaran pekerjaan inferensi Batch

Saat Anda membuat pekerjaan inferensi batch, job akan mengimpor data input batch Anda dari bucket Amazon S3, menggunakan versi solusi untuk menghasilkan rekomendasi item, dan mengekspor rekomendasi ke bucket Amazon S3 dalam format JSON.

Bagian berikut mencantumkan contoh file keluaran untuk pekerjaan inferensi batch menurut jenis resep. Anda tidak bisa mendapatkan rekomendasi batch dengan Trending-Now atau resep. Next-Best-Action

Topik

- Resep USER_PERSONALIZATION
- Resep POPULAR_ITEMS
- resep PERSONALIZED_RANKING
- Resep RELATED_ITEMS

Resep USER_PERSONALIZATION

Berikut ini adalah contoh file JSON output untuk resep USER_PERSONALIZATION.

```
{"input":{"userId":"4638"},"output":{"recommendedItems":
["63992","115149","110102","148626","148888","31685","102445","69526","92535","143355","62374",
[0.0152238,0.0069081,0.0068222,0.006394,0.0059746,0.0055851,0.0049357,0.0044644,0.0042968,0.004
{"input":{"userId":"663"},"output":{"recommendedItems":
["368","377","25","780","1610","648","1270","6","165","1196","1097","300","1183","608","104","4
[0.0406197,0.0372557,0.0254077,0.0151975,0.014991,0.0127175,0.0124547,0.0116712,0.0091098,0.008
```

```
{"input":{"userId":"3384"},"output":{"recommendedItems":
["597","21","223","2144","208","2424","594","595","920","104","520","367","2081","39","1035","2
[0.0241061,0.0119394,0.0118012,0.010662,0.0086972,0.0079428,0.0073218,0.0071438,0.0069602,0.005
...
```

Resep POPULAR_ITEMS

Contoh berikut menunjukkan format file JSON output untuk resep Popularity-Count. Anda tidak bisa mendapatkan rekomendasi batch dengan resep Trending-Now.

```
{"input": {"userId": "12"}, "output": {"recommendedItems": ["105", "106", "441"]}}
{"input": {"userId": "105"}, "output": {"recommendedItems": ["105", "106", "441"]}}
{"input": {"userId": "41"}, "output": {"recommendedItems": ["105", "106", "441"]}}
...
```

resep PERSONALIZED_RANKING

Contoh berikut menunjukkan format file JSON output untuk resep PERSONALIZED_RANKING.

```
{"input":{"userId":"891","itemList":["27","886","101"]},"output":{"recommendedItems":
["27","101","886"],"scores":[0.48421,0.28133,0.23446]}}
{"input":{"userId":"445","itemList":["527","55","901"]},"output":{"recommendedItems":
["901","527","55"],"scores":[0.46972,0.31011,0.22017]}}
{"input":{"userId":"71","itemList":["29","351","199"]},"output":{"recommendedItems":
["351","29","199"],"scores":[0.68937,0.24829,0.06232]}}
...
```

Resep RELATED_ITEMS

Contoh berikut menunjukkan format file JSON output untuk resep RELATED_ITEMS.

```
{"input": {"itemId": "105"}, "output": {"recommendedItems": ["106", "107", "49"]}}
{"input": {"itemId": "106"}, "output": {"recommendedItems": ["105", "107", "49"]}}
{"input": {"itemId": "441"}, "output": {"recommendedItems": ["2", "442", "435"]}}
...
```

Contoh berikut menunjukkan format file JSON output untuk resep Similar-Items dengan tema. Untuk informasi selengkapnya tentang rekomendasi dengan tema, lihat[Rekomendasi Batch dengan tema dari Content Generator](#).

```
{"input":{"itemId":"40"},"output":{"recommendedItems":
["36","50","44","22","21","29","3","1","2","39"],"theme":"Movies
 with a strong female lead","itemsThemeRelevanceScores":
[0.19994527,0.183059963,0.17478035,0.1618133,0.1574806,0.15468733,0.1499242,0.14353688,0.135314
{"input":{"itemId":"43"},"output":{"recommendedItems":
["50","21","36","3","17","2","39","1","10","5"],"theme":"The best movies of
 1995","itemsThemeRelevanceScores":
[0.184988,0.1795761,0.11143453,0.0989443,0.08258403,0.07952615,0.07115086,0.0621634,-0.138913,-
...
```


Mendapatkan segmen pengguna batch dengan sumber daya khusus

Untuk mendapatkan segmen pengguna, Anda menggunakan pekerjaan segmen batch. Pekerjaan segmen batch adalah alat yang mengimpor data masukan batch Anda dari bucket Amazon S3 dan menggunakan versi solusi yang dilatih dengan resep USER_SEGMENTATION untuk menghasilkan segmen pengguna untuk setiap baris data input.

Bergantung pada resepnya, data input adalah daftar item atau atribut metadata item dalam format JSON. Untuk atribut item, data input Anda dapat menyertakan ekspresi untuk membuat segmen pengguna berdasarkan beberapa atribut metadata. Pekerjaan segmen batch mengekspor segmen pengguna ke bucket Amazon S3 keluaran. Setiap segmen pengguna diurutkan dalam urutan menurun berdasarkan probabilitas bahwa setiap pengguna akan berinteraksi dengan item dalam data input Anda.

Saat membuat segmen pengguna, Amazon Personalize mempertimbangkan data dalam kumpulan data dari impor massal dan individual:

- Untuk data massal, Amazon Personalize menghasilkan segmen hanya menggunakan data massal yang ada pada pelatihan versi solusi lengkap terakhir. Dan itu hanya menggunakan data massal yang Anda impor dengan mode impor FULL (mengganti data yang ada).
- Untuk data dari operasi impor data individual, Amazon Personalize menghasilkan segmen pengguna menggunakan data yang ada pada pelatihan versi solusi lengkap terakhir. Agar catatan yang lebih baru memengaruhi segmen pengguna, buat versi solusi baru, lalu buat pekerjaan segmen batch.

Menghasilkan segmen pengguna berfungsi sebagai berikut:

1. Siapkan dan unggah data masukan Anda dalam format JSON ke bucket Amazon S3. Format data input Anda tergantung pada resep yang Anda gunakan dan pekerjaan yang Anda buat. Lihat Mempersiapkan data input untuk segmen pengguna.
2. Buat lokasi terpisah untuk data keluaran Anda, baik folder lain atau bucket Amazon S3 yang berbeda.
3. Buat pekerjaan segmen batch. Lihat Mendapatkan segmen pengguna dengan pekerjaan segmen batch.
4. Saat tugas segmen batch selesai, ambil segmen pengguna dari lokasi keluaran Anda di Amazon S3.

Topik

- Pedoman dan persyaratan untuk mendapatkan segmen pengguna
- Mempersiapkan data input untuk segmen pengguna
- Mendapatkan segmen pengguna dengan pekerjaan segmen batch
- Contoh format output pekerjaan segmen Batch

Pedoman dan persyaratan untuk mendapatkan segmen pengguna

Berikut ini adalah pedoman dan persyaratan untuk mendapatkan segmen batch:

- Anda harus menggunakan resep USER_SEGMENTATION.
- Peran layanan Amazon Personalisasi IAM Anda memerlukan izin untuk membaca dan menambahkan file ke bucket Amazon S3 Anda. Untuk informasi tentang pemberian izin, lihat. Kebijakan peran layanan untuk alur kerja batch Untuk informasi selengkapnya tentang izin bucket, lihat contoh kebijakan pengguna di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

  Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.
- Anda harus membuat solusi kustom dan versi solusi sebelum Anda membuat pekerjaan inferensi batch. Namun, Anda tidak perlu membuat kampanye Amazon Personalize. Jika Anda membuat grup kumpulan data Domain, Anda masih dapat membuat sumber daya kustom.

- Data masukan Anda harus diformat seperti yang dijelaskan dalamMempersiapkan data input untuk segmen pengguna.
- Jika Anda menggunakan Item-Attribute-Affinity resep, atribut dalam data input Anda tidak dapat menyertakan metadata item tekstual tidak terstruktur, seperti deskripsi produk.
- Jika Anda menggunakan filter dengan parameter placeholder, Anda harus menyertakan nilai untuk parameter dalam data input Anda dalam objek. `filterValues` Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda.
- Kami menyarankan Anda menggunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda) daripada data input Anda.

Mempersiapkan data input untuk segmen pengguna

Pekerjaan segmen batch menggunakan versi solusi untuk membuat segmen pengguna berdasarkan data yang Anda berikan dalam file JSON masukan. Sebelum Anda bisa mendapatkan segmen pengguna, Anda harus menyiapkan dan mengunggah file JSON Anda ke bucket Amazon S3. Sebaiknya buat folder keluaran di bucket Amazon S3 atau gunakan bucket Amazon S3 keluaran terpisah. Anda kemudian dapat menjalankan beberapa pekerjaan inferensi batch menggunakan lokasi data input yang sama.

Jika Anda menggunakan filter dengan parameter placeholder, seperti`$GENRE`, Anda harus memberikan nilai untuk parameter dalam `filterValues` objek di JSON input Anda. Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda.

Untuk menyiapkan dan mengimpor data

1. Format data input batch Anda tergantung pada resep yang digunakan solusi Anda. Pisahkan elemen data input dengan baris baru. Data masukan Anda adalah daftar ItemIds (Item-Affinity) atau atribut item (Item-Atribute-Affinity).

   - Untuk atribut item, data input dapat menyertakan ekspresi logis dengan `AND` operator untuk mendapatkan pengguna untuk beberapa item atau atribut per kueri. Untuk informasi selengkapnya, lihat Menentukan atribut item untuk resep Item-Attribute-Affinity.
   - Untuk atribut item, gunakan \ karakter untuk menghindari karakter khusus dan tanda kutip tunggal atau ganda dalam data input Anda.
   - Untuk contoh data masukan untuk kedua resep, lihatContoh JSON masukan dan keluaran pekerjaan segmen Batch.

2. Unggah JSON masukan Anda ke folder input di bucket Amazon S3 Anda. Untuk informasi selengkapnya, lihat Mengunggah file dan folder menggunakan seret dan lepas di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon
3. Buat lokasi terpisah untuk data keluaran Anda, baik folder atau bucket Amazon S3 lainnya. Dengan membuat lokasi terpisah untuk JSON keluaran, Anda dapat menjalankan beberapa pekerjaan segmen batch dengan lokasi data input yang sama.

Setelah menyiapkan data input dan mengunggahnya ke bucket Amazon S3, Anda siap menghasilkan segmen pengguna dengan pekerjaan segmen batch. Untuk informasi selengkapnya, lihat Mendapatkan segmen pengguna dengan pekerjaan segmen batch.

Topik

- Menentukan atribut item untuk resep Item-Attribute-Affinity
- Contoh JSON masukan dan keluaran pekerjaan segmen Batch

Menentukan atribut item untuk resep Item-Attribute-Affinity

Jika Anda menggunakan Item-Attribute-Affinity resep, data input Anda adalah daftar atribut item. Anda dapat mencampur kolom metadata yang berbeda. Misalnya satu baris mungkin kolom numerik dan yang berikutnya mungkin kolom kategoris. Anda tidak dapat menggunakan metadata item tekstual tidak terstruktur sebagai atribut item.

Metadata item masukan Anda dapat menyertakan ekspresi logis dengan `AND` operator untuk mendapatkan segmen pengguna untuk beberapa atribut. Misalnya, baris data input Anda mungkin `{"itemAttributes": "ITEMS.genres = \"Comedy\" AND ITEMS.genres = \"Action \""}` atau`{"itemAttributes": "ITEMS.genres = "\Comedy\" AND ITEMS.audience = "\teen\""}`.

Saat menggabungkan dua atribut dengan `AND` operator, Anda membuat segmen pengguna dengan pengguna yang lebih cenderung berinteraksi dengan item yang memiliki kedua atribut berdasarkan riwayat interaksi pengguna. Tidak seperti ekspresi filter (yang menggunakan `IN` operator untuk kesetaraan string), ekspresi input segmen batch hanya mendukung = simbol untuk kesetaraan untuk pencocokan string.

Contoh JSON masukan dan keluaran pekerjaan segmen Batch

Untuk pekerjaan segmen batch, data input Anda harus berupa daftar ItemIds (resep Item-Affinity) atau atribut item (Item-Atribute-Affinity). Setiap baris data input adalah kueri inferensi terpisah. Setiap segmen pengguna diurutkan dalam urutan menurun berdasarkan probabilitas bahwa setiap pengguna akan berinteraksi dengan item dalam inventaris Anda.

Jika Anda menggunakan filter dengan parameter placeholder, seperti`$GENRE`, Anda harus memberikan nilai untuk parameter dalam `filterValues` objek di JSON input Anda. Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda.

Berikut ini adalah contoh input dan output JSON yang diformat dengan benar untuk pekerjaan segmen batch yang diatur berdasarkan resep.

Barang-Afinitas

Input

Data input Anda dapat memiliki maksimal 500 item. Pisahkan masing-masing `itemId` dengan baris baru sebagai berikut.

```
{"itemId": "105"}
{"itemId": "106"}
{"itemId": "441"}
...
```

Output

```
{"input": {"itemId": "105"}, "output": {"recommendedUsers": ["106", "107", "49"]}}
{"input": {"itemId": "106"}, "output": {"recommendedUsers": ["105", "107", "49"]}}
{"input": {"itemId": "441"}, "output": {"recommendedUsers": ["2", "442", "435"]}}
...
```

Item-Atribusi-Afinitas

Input

Data masukan Anda dapat memiliki maksimal 10 kueri, di mana setiap kueri adalah satu atau lebih atribut item non-tekstual. Pisahkan setiap atribut atau atribut ekspresi dengan baris baru sebagai berikut.

```
{"itemAttributes": "ITEMS.genres = \"Comedy\" AND ITEMS.genres = \"Action\""}
{"itemAttributes": "ITEMS.genres = \"Comedy\""}
{"itemAttributes": "ITEMS.genres = \"Horror\" AND ITEMS.genres = \"Action\""}
...
```

Output

```
{"itemAttributes": "ITEMS.genres = \"Comedy\" AND ITEMS.genres = \"Action\"",
 "output": {"recommendedUsers": ["25", "78", "108"]}}
{"itemAttributes": "ITEMS.genres = \"Adventure\"", "output": {"recommendedUsers":
 ["87", "31", "129"]}}
{"itemAttributes": "ITEMS.genres = \"Horror\" AND ITEMS.genres = \"Action\"",
 "output": {"recommendedUsers": ["8", "442", "435"]}}
...
```

Mendapatkan segmen pengguna dengan pekerjaan segmen batch

Jika Anda menggunakan resep USER_SEGMENTATION, Anda dapat membuat pekerjaan segmen batch untuk mendapatkan segmen pengguna dengan versi solusi Anda. Setiap segmen pengguna diurutkan dalam urutan menurun berdasarkan probabilitas bahwa setiap pengguna akan berinteraksi dengan item dalam inventaris Anda. Bergantung pada resepnya, data masukan Anda harus berupa daftar item (Resep item-Affinity) atau atribut item (Item-Attribute-Affinity resep) dalam format JSON. Anda dapat membuat pekerjaan segmen batch dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs

Saat membuat pekerjaan segmen batch, Anda menentukan jalur Amazon S3 ke lokasi input dan output Anda. Amazon S3 berbasis awalan. Jika Anda memberikan awalan untuk lokasi data input, Amazon Personalize menggunakan semua file yang cocok dengan awalan tersebut sebagai data input. Misalnya, jika Anda menyediakan `s3://amzn-s3-demo-bucket/folderName` dan bucket Anda juga memiliki folder dengan jalur`s3://amzn-s3-demo-bucket/folderName_test`, Amazon Personalize menggunakan semua file di kedua folder sebagai data input. Untuk hanya menggunakan file dalam folder tertentu sebagai data input, akhiri jalur Amazon S3 dengan pembatas awalan, seperti/: `s3://amzn-s3-demo-bucket/folderName/` Untuk informasi selengkapnya tentang cara Amazon S3 mengatur objek, lihat Mengatur, mencantumkan, dan bekerja dengan objek Anda.

Topik

- Membuat pekerjaan segmen batch (konsol)

- Membuat pekerjaan segmen batch (AWS CLI)
- Membuat pekerjaan segmen batch (AWS SDKs)

Membuat pekerjaan segmen batch (konsol)

Setelah selesaiMempersiapkan data input untuk rekomendasi batch, Anda siap untuk membuat pekerjaan segmen batch. Prosedur ini mengasumsikan bahwa Anda telah membuat solusi dan versi solusi (model terlatih) dengan resep USER_SEGEMENTATION.

Untuk mendapatkan buat pekerjaan segmen batch (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Datasets, pilih grup dataset Anda.
3. Pilih pekerjaan segmen batch di panel navigasi, lalu pilih Buat pekerjaan segmen batch.
4. Dalam detail pekerjaan segmen batch, untuk nama pekerjaan segmen Batch, tentukan nama untuk pekerjaan segmen batch Anda.
5. Untuk Solusi, pilih solusi dan kemudian pilih ID versi Solusi yang ingin Anda gunakan untuk menghasilkan rekomendasi. Anda dapat membuat pekerjaan segmen batch hanya jika Anda menggunakan resep USER_SEGEMENTATION.
6. Untuk Jumlah pengguna, secara opsional menentukan jumlah pengguna Amazon Personalize menghasilkan untuk setiap segmen pengguna. Default-nya adalah 25. Maksimal 5 juta.
7. Untuk sumber Input, tentukan jalur Amazon S3 ke file input Anda atau gunakan Browse S3 untuk memilih bucket Amazon S3 Anda.

   Gunakan sintaks berikut: **`s3://amzn-s3-demo-bucket/<folder name>/<input JSON file name>.json`**

   Data masukan Anda harus dalam format yang benar untuk resep yang digunakan solusi Anda. Untuk contoh data masukan lihatContoh JSON masukan dan keluaran pekerjaan segmen Batch.
8. Untuk tujuan Output, tentukan jalur ke lokasi keluaran Anda atau gunakan Browse S3 untuk memilih bucket Amazon S3 Anda. Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda).

   Gunakan sintaks berikut: **`s3://amzn-s3-demo-bucket/<output folder name>/`**
9. Untuk peran IAM, pilih salah satu dari berikut ini:

- Pilih Buat dan gunakan peran layanan baru dan masukkan nama peran Layanan untuk membuat peran baru, atau
- Jika Anda telah membuat peran dengan izin yang benar, pilih Gunakan peran layanan yang ada dan pilih peran IAM.

Peran yang Anda gunakan harus memiliki akses baca dan tulis ke bucket Amazon S3 masukan dan keluaran Anda masing-masing.

10. Untuk konfigurasi Filter opsional pilih filter untuk menerapkan filter ke segmen pengguna. Jika filter Anda menggunakan parameter placeholder, pastikan nilai untuk parameter disertakan dalam input JSON Anda. Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda.
11. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya
12. Pilih Buat pekerjaan segmen batch. Pembuatan pekerjaan segmen Batch dimulai dan halaman pekerjaan segmen Batch muncul dengan bagian detail pekerjaan segmen Batch ditampilkan.
13. Jika status pekerjaan segmen batch berubah menjadi Aktif, Anda dapat mengambil output pekerjaan dari bucket Amazon S3 keluaran yang ditentukan. Nama file output akan menjadi format*`input-name`*`.out`.

Membuat pekerjaan segmen batch (AWS CLI)

Setelah selesaiMempersiapkan data input untuk rekomendasi batch, Anda siap untuk membuat pekerjaan segmen batch menggunakan `create-batch-segment-job` kode berikut. Tentukan nama pekerjaan, ganti `Solution version ARN` dengan Amazon Resource Name (ARN) versi solusi Anda, dan ganti `IAM service role ARN` dengan ARN peran layanan IAM yang Anda buat untuk Amazon Personalize selama penyiapan. Peran ini harus memiliki akses baca dan tulis ke bucket Amazon S3 masukan dan keluaran Anda masing-masing. Untuk `num-results` tentukan jumlah pengguna yang ingin diprediksi Amazon Personalize untuk setiap baris data masukan. Default-nya adalah 25. Maksimal 5 juta. Secara opsional menyediakan `filter-arn` untuk memfilter segmen pengguna. Jika filter Anda menggunakan parameter placeholder, pastikan nilai untuk parameter disertakan dalam input JSON Anda. Untuk informasi selengkapnya, lihat Memfilter rekomendasi batch dan segmen pengguna (sumber daya khusus).

Ganti `S3 input path` dan `S3 output path` dengan jalur Amazon S3 ke file input dan lokasi output Anda. Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda). Gunakan sintaks berikut untuk lokasi input dan output: **`s3://amzn-s3-demo-bucket/<folder name>/<input JSON file name>.json`** dan**`s3://amzn-s3-demo-bucket/<output folder name>/`**.

```
aws personalize create-batch-segment-job \
                --job-name Job name \
                --solution-version-arn Solution version ARN \
                --num-results The number of predicted users \
                --filter-arn Filter ARN \
                --job-input s3DataSource={path=s3://S3 input path} \
                --job-output s3DataDestination={path=s3://S3 output path} \
                --role-arn IAM service role ARN
{
   "batchSegmentJobArn": "arn:aws:personalize:us-west-2:acct-id:batch-segment-job/
batchSegmentJobName"
}
```

Membuat pekerjaan segmen batch (AWS SDKs)

Setelah selesaiMempersiapkan data input untuk rekomendasi batch, Anda siap untuk membuat pekerjaan segmen batch dengan `CreateBatchSegmentJob` operasi. Kode berikut menunjukkan cara membuat pekerjaan segmen batch. Beri nama pekerjaan, tentukan Nama Sumber Daya Amazon (ARN) versi solusi yang akan digunakan, tentukan ARN untuk peran Amazon Personalisasi IAM, dan tentukan jalur Amazon S3 ke file input dan lokasi keluaran Anda. Peran layanan IAM Anda harus memiliki akses baca dan tulis ke bucket Amazon S3 masukan dan keluaran Anda masing-masing.

Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda). Gunakan sintaks berikut untuk lokasi input dan output: **`s3://amzn-s3-demo-bucket/<folder name>/<input JSON file name>.json`** dan**`s3://amzn-s3-demo-bucket/<output folder name>/`**.

Untuk`numResults`, tentukan jumlah pengguna yang ingin diprediksi Amazon Personalize untuk setiap baris data input. Default-nya adalah 25. Maksimal 5 juta. Secara opsional menyediakan `filterArn` untuk memfilter segmen pengguna. Jika filter Anda menggunakan parameter placeholder, pastikan nilai untuk parameter disertakan dalam input JSON Anda. Untuk informasi selengkapnya, lihat Memfilter rekomendasi batch dan segmen pengguna (sumber daya khusus).

SDK for Python (Boto3)

```
import boto3

personalize_rec = boto3.client(service_name='personalize')

personalize_rec.create_batch_segment_job (
    solutionVersionArn = "Solution version ARN",
    jobName = "Job name",
    numResults = 25,
    filterArn = "Filter ARN",
    roleArn = "IAM service role ARN",
    jobInput =
       {"s3DataSource": {"path": "s3://amzn-s3-demo-bucket/<folder name>/<input JSON
 file name>.json"}},
    jobOutput =
       {"s3DataDestination": {"path": "s3://amzn-s3-demo-bucket/<output folder
 name>/"}}
)
```

SDK for Java 2.x

```
public static String createBatchSegmentJob(PersonalizeClient personalizeClient,
                                                        String solutionVersionArn,
                                                        String jobName,
                                                        String filterArn,
                                                        int numResults,
                                                        String
 s3InputDataSourcePath,
                                                        String
 s3DataDestinationPath,
                                                        String roleArn,
                                                        String explorationWeight,
                                                        String
 explorationItemAgeCutOff) {

  long waitInMilliseconds = 60 * 1000;
  String status;
  String batchSegmentJobArn;

  try {
      // Set up data input and output parameters.
      S3DataConfig inputSource = S3DataConfig.builder()
```

```
                .path(s3InputDataSourcePath)
                .build();
        S3DataConfig outputDestination = S3DataConfig.builder()
                .path(s3DataDestinationPath)
                .build();

        BatchSegmentJobInput jobInput = BatchSegmentJobInput.builder()
                .s3DataSource(inputSource)
                .build();
        BatchSegmentJobOutput jobOutputLocation = BatchSegmentJobOutput.builder()
                .s3DataDestination(outputDestination)
                .build();


        CreateBatchSegmentJobRequest createBatchSegmentJobRequest =
CreateBatchSegmentJobRequest.builder()
                .solutionVersionArn(solutionVersionArn)
                .filterArn(filterArn)
                .jobInput(jobInput)
                .jobOutput(jobOutputLocation)
                .jobName(jobName)
                .numResults(numResults)
                .roleArn(roleArn)
                .build();

        batchSegmentJobArn =
personalizeClient.createBatchSegmentJob(createBatchSegmentJobRequest)
                .batchSegmentJobArn();
        DescribeBatchSegmentJobRequest describeBatchSegmentJobRequest =
DescribeBatchSegmentJobRequest.builder()
                .batchSegmentJobArn(batchSegmentJobArn)
                .build();

        long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

        // wait until the batch segment job is complete.
        while (Instant.now().getEpochSecond() < maxTime) {

            BatchSegmentJob batchSegmentJob = personalizeClient
                    .describeBatchSegmentJob(describeBatchSegmentJobRequest)
                    .batchSegmentJob();

            status = batchSegmentJob.status();
            System.out.println("batch segment job status: " + status);
```

```
            if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
                break;
            }
            try {
                Thread.sleep(waitInMilliseconds);
            } catch (InterruptedException e) {
                System.out.println(e.getMessage());
            }
        }
        return batchSegmentJobArn;

    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateBatchSegmentJobCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the batch segment job's parameters.

export const createBatchSegmentJobParam = {
  jobName: "NAME",
  jobInput: {
    s3DataSource: {
      path: "INPUT_PATH",
    },
  },
  jobOutput: {
    s3DataDestination: {
      path: "OUTPUT_PATH",
    },
  },
  roleArn: "ROLE_ARN",
  solutionVersionArn: "SOLUTION_VERSION_ARN",
```

```
  numResults: 20,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateBatchSegmentJobCommand(createBatchSegmentJobParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Memproses pekerjaan batch mungkin membutuhkan waktu beberapa saat untuk diselesaikan. Anda dapat memeriksa status pekerjaan dengan memanggil DescribeBatchSegmentJob dan meneruskan `batchSegmentJobArn` sebagai parameter input. Anda juga dapat mencantumkan semua pekerjaan segmen batch Amazon Personalisasi di AWS lingkungan Anda dengan menelepon. ListBatchSegmentJobs

Contoh format output pekerjaan segmen Batch

Pekerjaan segmen batch mengimpor data input batch Anda dari bucket Amazon S3, menggunakan versi solusi yang dilatih dengan resep USER_SEGMENTATION untuk menghasilkan segmen pengguna, dan mengekspor segmen ke bucket Amazon S3.

Bagian berikut mencantumkan contoh keluaran JSON untuk pekerjaan segmen batch berdasarkan resep.

Topik

- Barang-Afinitas
- Item-Atribusi-Afinitas

Barang-Afinitas

Contoh berikut menunjukkan format file JSON output untuk resep Item-Affinity.

```
{"input": {"itemId": "105"}, "output": {"recommendedUsers": ["106", "107", "49"]}}
{"input": {"itemId": "106"}, "output": {"recommendedUsers": ["105", "107", "49"]}}
{"input": {"itemId": "441"}, "output": {"recommendedUsers": ["2", "442", "435"]}}
...
```

Item-Atribusi-Afinitas

Contoh berikut menunjukkan format file JSON output untuk Item-Attribute-Affinity resep.

```
{"itemAttributes": "ITEMS.genres = \"Comedy\" AND ITEMS.genres = \"Action\"", "output":
 {"recommendedUsers": ["25", "78", "108"]}}
{"itemAttributes": "ITEMS.genres = \"Adventure\"", "output": {"recommendedUsers":
 ["87", "31", "129"]}}
{"itemAttributes": "ITEMS.genres = \"Horror\" AND ITEMS.genres = \"Action\"", "output":
 {"recommendedUsers": ["8", "442", "435"]}}
...
```

Memfilter rekomendasi dan segmen pengguna

Saat mendapatkan rekomendasi dengan pemberi rekomendasi domain atau kampanye khusus, Anda dapat memfilter hasil berdasarkan kriteria khusus. Misalnya, Anda mungkin tidak ingin merekomendasikan produk yang telah dibeli pengguna atau hanya merekomendasikan item untuk kelompok usia tertentu.

Demikian pula, dengan resep USER_SEGMENTATION, Anda mungkin tidak ingin menyertakan jenis pengguna tertentu di segmen pengguna. Dengan memfilter hasil Anda, Anda dapat mengontrol item yang akan direkomendasikan kepada pengguna atau pengguna yang akan disertakan dalam segmen pengguna.

Anda dapat membuat, mengedit, menghapus, dan menerapkan filter menggunakan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), dan. AWS SDKs

- Untuk rekomendasi waktu nyata, Anda menerapkan filter dan menentukan nilai parameter filter apa pun saat Anda memanggil GetRecommendations, GetActionRecommendations, atau GetPersonalizedRanking operasi. Anda juga dapat menerapkan filter ketika Anda mendapatkan rekomendasi dari kampanye atau pemberi rekomendasi di konsol.

  Saat Anda mendapatkan rekomendasi item waktu nyata dengan resep atau kasus penggunaan item yang dipersonalisasi atau terkait, Anda dapat menentukan promosi dalam permintaan Anda. Promosi menggunakan filter untuk menentukan aturan bisnis tambahan yang berlaku untuk subset item yang direkomendasikan yang dapat dikonfigurasi. Untuk informasi selengkapnya, lihat Mempromosikan item dalam rekomendasi waktu nyata.
- Untuk alur kerja batch, Anda menyertakan nilai parameter filter apa pun di JSON masukan Anda. Kemudian Anda menentukan Nama Sumber Daya Amazon (ARN) filter saat Anda membuat pekerjaan inferensi batch atau pekerjaan segmen batch. Untuk informasi selengkapnya, lihat Memfilter rekomendasi batch dan segmen pengguna (sumber daya khusus).

Filter pembaruan untuk catatan baru

Untuk data yang Anda impor dengan PutEvents atau PutActionInteractions operasi, Amazon Personalize memperbarui filter apa pun dalam grup kumpulan data dengan data baru dalam hitungan detik setelah impor. Misalnya, jika filter Anda menghapus item yang dibeli dari rekomendasi, dan Anda merekam peristiwa pembelian untuk pengguna dengan PutEvents operasi tersebut, item ini

akan dihapus dari rekomendasi future untuk pengguna ini dalam hitungan detik setelah merekam acara.

Untuk semua data lain yang diimpor secara massal atau individual, Amazon Personalize memperbarui filter apa pun dalam grup kumpulan data dengan data baru dalam waktu 20 menit dari impor terakhir.

Topik

- Ekspresi Filter
- Memfilter rekomendasi waktu nyata
- Memfilter rekomendasi batch dan segmen pengguna (sumber daya khusus)

Ekspresi Filter

Untuk mengonfigurasi filter, Anda harus menggunakan ekspresi filter yang diformat dengan benar. Ekspresi filter terdiri dari kumpulan data dan pengidentifikasi bidang di. `dataset fieldformat`, bersama dengan operator logis, kata kunci, dan nilai. Untuk nilai, Anda dapat menentukan nilai tetap atau menambahkan parameter placeholder mengatur kriteria filter saat Anda mendapatkan rekomendasi.

Anda dapat menggunakan ekspresi filter untuk memfilter item, pengguna, atau tindakan dari rekomendasi berdasarkan data dari kumpulan data berikut:

- Interaksi item: Anda dapat menggunakan ekspresi filter untuk menyertakan atau mengecualikan item atau pengguna berdasarkan data interaksi. Misalnya, Anda dapat mengecualikan item yang telah diklik pengguna (untuk rekomendasi item), atau hanya menyertakan pengguna yang telah menilai item (untuk resep Item-Affinity). Untuk semua jenis resep, Anda dapat memfilter hanya berdasarkan jenis acara. Anda tidak dapat memfilter berdasarkan metadata interaksi lain, seperti metadata kontekstual. Anda tidak dapat menggunakan filter interaksi item dengan fileItem-Attribute-Affinity resep.

  Amazon Personalize mempertimbangkan hingga 100 interaksi terbaru per pengguna per jenis acara. Ini adalah kuota yang dapat disesuaikan. Anda dapat meminta peningkatan kuota menggunakan konsol Service Quotas. Jika Anda tidak mengimpor interaksi item untuk pengguna selama tiga bulan, filter Anda tidak lagi mempertimbangkan data historis pengguna. Untuk mempertimbangkan data ini, Anda harus mengimpor seluruh riwayat peristiwa pengguna lagi.

- Interaksi tindakan: Gunakan ekspresi filter untuk menyertakan atau mengecualikan tindakan yang telah berinteraksi dengan pengguna berdasarkan jenis peristiwa. Misalnya, Anda mungkin mengecualikan tindakan yang telah dilakukan pengguna. Anda tidak dapat memfilter berdasarkan metadata interaksi tindakan lainnya.

  Amazon Personalize mempertimbangkan hingga 300 interaksi tindakan terbaru per pengguna per jenis peristiwa. Ini adalah kuota yang dapat disesuaikan. Anda dapat meminta peningkatan kuota menggunakan konsol Service Quotas.
- Item: Gunakan ekspresi filter untuk menyertakan atau mengecualikan item berdasarkan kondisi item tertentu. Anda tidak dapat menggunakan filter untuk menyertakan atau mengecualikan item berdasarkan metadata item tekstual tidak terstruktur seperti deskripsi produk. Jika kasus penggunaan domain atau resep kustom Anda menghasilkan rekomendasi item terkait, seperti resep Item Serupa atau kasus penggunaan domain More Like X, Anda dapat menggunakan ekspresi filter untuk menyertakan atau mengecualikan item berdasarkan properti item yang Anda tentukan dalam permintaan rekomendasi.
- Pengguna: Untuk rekomendasi item dan tindakan, jika Anda memiliki kumpulan data Pengguna, Anda dapat mengecualikan atau menyertakan item atau tindakan berdasarkan data. `CurrentUser` Untuk rekomendasi yang dipersonalisasi, item populer, dan rekomendasi tindakan, ini adalah pengguna yang Anda dapatkan rekomendasinya. Untuk item terkait, ini pengguna opsional yang dapat Anda tentukan dalam permintaan rekomendasi Anda.

  Untuk segmen pengguna, Anda dapat menggunakan ekspresi filter untuk menyertakan atau mengecualikan pengguna dari segmen pengguna berdasarkan atribut, seperti`Users.MEMBERSHIP_STATUS`.
- Tindakan: Gunakan ekspresi filter untuk menyertakan atau mengecualikan tindakan berdasarkan kondisi tindakan tertentu. Amazon Personalize secara otomatis mengecualikan tindakan berdasarkan `Action expiration timestamp` dan data. `Repeat frequency` Anda tidak dapat membuat filter khusus tambahan yang memfilter berdasarkan data ini.

Untuk daftar lengkap elemen ekspresi filter, lihatFilter elemen ekspresi. Untuk contoh ekspresi filter, lihatContoh ekspresi filter.

Topik

- Pedoman dan persyaratan
- Filter struktur ekspresi dan elemen
- Contoh ekspresi filter

Pedoman dan persyaratan

Saat membuat ekspresi filter, perhatikan pedoman dan persyaratan berikut:

- Anda tidak dapat menggunakan filter untuk menyertakan atau mengecualikan item berdasarkan metadata item tekstual tidak terstruktur seperti deskripsi produk.
- Jika Anda memfilter berdasarkan data interaksi item atau tindakan, Anda hanya dapat memfilter berdasarkan jenis acara. Anda tidak dapat memfilter berdasarkan metadata interaksi lain, seperti metadata kontekstual.
- Amazon Personalize mengabaikan case hanya saat mencocokkan jenis peristiwa.
- Anda tidak dapat menggunakan Interaksi Item dan kumpulan data Item dalam satu ekspresi. Untuk membuat filter yang memfilter berdasarkan Interaksi dan kemudian kumpulan data Item (atau sebaliknya), Anda harus menghubungkan dua atau lebih ekspresi bersama-sama. Untuk informasi selengkapnya, lihat Menggabungkan beberapa ekspresi.
- Anda tidak dapat menggunakan interaksi Item dan kumpulan data Tindakan dalam satu ekspresi. Untuk membuat filter yang memfilter berdasarkan interaksi Item dan kemudian kumpulan data Tindakan (atau sebaliknya), Anda harus menghubungkan dua atau lebih ekspresi bersama-sama. Untuk informasi selengkapnya, lihat Menggabungkan beberapa ekspresi.
- Anda tidak dapat menggunakan filter interaksi item dengan fileItem-Attribute-Affinity resep.
- Anda tidak dapat membuat ekspresi filter yang memfilter menggunakan nilai dengan tipe boolean dalam skema Anda. Untuk memfilter berdasarkan nilai boolean, gunakan skema dengan bidang tipe String dan gunakan nilai `"True"` dan data `"False"` Anda. Atau Anda dapat menggunakan tipe int atau panjang dan nilai `0` dan`1`.
- Jumlah maksimum bidang kumpulan data yang berbeda untuk filter, baik dalam satu ekspresi atau di beberapa ekspresi yang dirantai bersama, adalah 10. Jumlah maksimum bidang kumpulan data yang berbeda di semua filter dalam grup kumpulan data adalah 20.
- Anda dapat menerapkan filter dengan CurrentItem elemen hanya jika kasus penggunaan domain atau resep khusus Anda menghasilkan rekomendasi item terkait, seperti resep Item Serupa atau kasus penggunaan domain More Like X.
- Anda tidak dapat menggunakan parameter placeholder dalam ekspresi filter yang menggunakan operator NOT_IN. Sebagai gantinya, gunakan operator IN dan gunakan Tindakan yang berlawanan. Misalnya, gunakan Sertakan alih-alih Kecualikan (atau sebaliknya).
- Anda tidak dapat membuat filter yang memfilter berdasarkan `Repeat frequency` data `Action expiration timestamp` dan. Amazon Personalize secara otomatis memfilter rekomendasi tindakan berdasarkan data ini.

Filter struktur ekspresi dan elemen

Bagian ini mencakup informasi tentang struktur ekspresi filter dan elemennya.

Topik

- Struktur ekspresi filter
- Filter elemen ekspresi

Struktur ekspresi filter

Struktur umum ekspresi filter adalah sebagai berikut:

```
EXCLUDE/INCLUDE ItemID/ActionID/UserID WHERE dataset type.field IN/NOT IN (value/
parameter)
```

Anda dapat membuat ekspresi filter secara manual atau mendapatkan bantuan dengan sintaks dan struktur ekspresi dengan menggunakan pembuat Ekspresi di konsol.

Filter elemen ekspresi

Gunakan elemen berikut untuk membuat ekspresi filter:

TERMASUK atau MENGECUALIKAN

Gunakan `INCLUDE` untuk membatasi rekomendasi hanya item yang memenuhi kriteria filter ATAU gunakan `EXCLUDE` untuk menghapus semua item yang memenuhi kriteria filter.

ItemID/ActionID/UserID

Gunakan salah satu elemen ini setelah `EXCLUDE` elemen `INCLUDE` atau. Elemen yang Anda gunakan bergantung pada apakah Anda memfilter item (untuk rekomendasi item), tindakan (untuk rekomendasi tindakan), atau pengguna (untuk segmen pengguna).

WHERE

Gunakan `WHERE` untuk memeriksa kondisi untuk item, tindakan, atau pengguna. Anda harus menggunakan `WHERE` elemen setelah`ItemID`,`ActionID`, atau`UserID`.

DAN/ATAU

Untuk menghubungkan beberapa kondisi bersama-sama dalam ekspresi filter yang sama, gunakan `AND` atau`OR`. Kondisi dirantai bersama menggunakan `AND` atau hanya `OR` dapat memengaruhi bidang kumpulan data yang digunakan dalam kondisi pertama.

Dataset.field

Berikan kumpulan data dan bidang metadata yang ingin Anda filter rekomendasi di. `dataset field`format. Misalnya, untuk memfilter rekomendasi item berdasarkan bidang genre di kumpulan data Item Anda, Anda akan menggunakan Items.genres dalam ekspresi filter Anda.

Kondisi IF

Gunakan `IF` kondisi hanya untuk memeriksa kondisi untuk `CurrentUser` dan hanya sekali di akhir ekspresi. Namun, Anda dapat memperpanjang `IF` kondisi menggunakan`AND`.

CurrentUser.atribut

Untuk memfilter rekomendasi item berdasarkan pengguna yang Anda dapatkan rekomendasinya, hanya dalam kondisi IF, gunakan `CurrentUser` dan berikan bidang pengguna. Misalnya, `CurrentUser.AGE`.

CurrentItem.atribut

Hanya untuk item terkait resep dan kasus penggunaan, gunakan`CurrentItem`. `attribute`untuk memfilter item berdasarkan atribut item yang Anda tentukan dalam permintaan Anda untuk rekomendasi item terkait. Misalnya, `CurrentItem.GENRE` atau `CurrentItem.PRICE`.

Anda dapat menerapkan filter dengan CurrentItem elemen hanya jika kasus penggunaan domain atau resep khusus Anda menghasilkan rekomendasi item terkait, seperti resep Item Serupa atau kasus penggunaan domain More Like X. Pertama kali Anda membuat filter dengan `CurrentItem` elemen, pembuatan filter bisa beberapa menit. Jika Anda menggunakan AWS KMS enkripsi, pembuatan filter dapat memakan waktu hingga 15 menit.

DALAM/TIDAK DI

Gunakan `IN` atau `NOT IN` sebagai operator perbandingan untuk memfilter berdasarkan pencocokan (atau tidak cocok) satu atau lebih nilai string. Amazon Personalisasi filter hanya pada string yang tepat.

Operator perbandingan

Gunakan =, <, <=, >, >=, dan! = operator untuk menguji data numerik, termasuk data yang diteruskan dalam parameter placeholder, untuk kesetaraan.

Karakter tanda bintang (*)

Gunakan * untuk memasukkan atau mengecualikan interaksi dari semua jenis. Gunakan * hanya untuk ekspresi filter yang menggunakan `EVENT_TYPE` bidang `Interactions` dataset.

Pemisah pipa

Gunakan pemisah pipa (|) untuk menyatukan beberapa ekspresi. Untuk informasi selengkapnya, lihat Menggabungkan beberapa ekspresi.

Parameter

Untuk ekspresi yang menggunakan operator perbandingan atau `IN` operator, gunakan tanda dolar ($) dan nama parameter untuk menambahkan parameter placeholder sebagai nilai. Misalnya, `$GENRES`. Misalnya, ketika Anda mendapatkan rekomendasi, Anda menyediakan genre atau genre untuk difilter.

> **Note**
>
> Anda menentukan nama parameter saat Anda menambahkannya ke ekspresi. Nama parameter tidak harus cocok dengan nama bidang. Kami menyarankan Anda menggunakan nama parameter yang mirip dengan nama bidang dan mudah diingat. Anda menggunakan nama parameter (case sensitive) saat menerapkan filter ke permintaan rekomendasi. Untuk contoh yang menunjukkan cara menerapkan filter dengan parameter placeholder saat menggunakan AWS SDKS, lihat. Menerapkan filter (AWS SDKs)

Contoh ekspresi filter

Gunakan ekspresi filter di bagian berikut untuk mempelajari cara membuat ekspresi filter Anda sendiri.

Topik

- Contoh ekspresi filter rekomendasi item
- Ekspresi filter segmen pengguna
- Contoh ekspresi filter rekomendasi tindakan

- Menggabungkan beberapa ekspresi

Contoh ekspresi filter rekomendasi item

Ekspresi filter berikut menunjukkan cara memfilter rekomendasi item berdasarkan interaksi item, metadata item, dan metadata pengguna. Mereka diatur oleh tipe data.

Topik

- Data interaksi item
- Data barang
- Data pengguna

Data interaksi item

Ekspresi berikut mengecualikan item berdasarkan jenis peristiwa (seperti klik) atau jenis peristiwa yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan `$EVENT_TYPE` parameter.

```
EXCLUDE ItemID WHERE Interactions.EVENT_TYPE IN ($EVENT_TYPE)
```

Ekspresi berikut mengecualikan item yang diklik atau dialirkan pengguna.

```
EXCLUDE ItemID WHERE Interactions.EVENT_TYPE IN ("click", "stream")
```

Ekspresi berikut hanya mencakup item yang telah diklik pengguna.

```
INCLUDE ItemID WHERE Interactions.EVENT_TYPE IN ("click")
```

Data barang

Ekspresi berikut mengecualikan item berdasarkan kategori atau kategori yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan `$CATEGORY` parameter.

```
EXCLUDE ItemID WHERE Items.CATEGORY IN ($CATEGORY)
```

Ekspresi berikut hanya mencakup item yang lebih murah daripada item saat ini (item yang Anda tentukan dalam permintaan rekomendasi item terkait), dan dibuat oleh studio yang sama dengan item saat ini. Anda dapat menerapkan filter dengan CurrentItem elemen hanya jika kasus penggunaan domain atau resep khusus Anda menghasilkan rekomendasi item terkait.

```
INCLUDE ItemID WHERE Items.PRICE < CurrentItem.PRICE AND Items.GENRE IN
 CurrentItem.GENRE
```

Ekspresi berikut mengecualikan item berdasarkan beberapa tingkat bidang kategoris. Ini mengecualikan item dengan nilai CATEGORY_L1 yang tidak memiliki nilai CATEGORY_L2 dari`shoe`. `boot`

```
EXCLUDE ItemID WHERE Items.CATEGORY_L1 IN ("shoe") AND Items.CATEGORY_L2 NOT IN
 ("boot")
```

Ekspresi berikut hanya mencakup item dengan harga kurang dari atau sama dengan harga yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan `$PRICE` parameter.

```
INCLUDE ItemID WHERE Items.PRICE <= $PRICE
```

Ekspresi berikut hanya mencakup item yang telah dibuat lebih awal dari stempel waktu (dalam waktu waktu Unix) yang Anda tentukan saat Anda mendapatkan rekomendasi.

```
INCLUDE ItemID WHERE Items.CREATION_TIMESTAMP < $DATE
```

Ekspresi berikut hanya mencakup item dengan genre atau genre yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan `$GENRE` parameter.

```
INCLUDE ItemID WHERE Items.GENRE IN ($GENRE)
```

Ekspresi berikut hanya mencakup item yang lebih mahal daripada item saat ini dan dibuat lebih baru daripada stempel waktu (dalam waktu epoch Unix) yang Anda tentukan. Anda dapat menggunakan filter ini jika Anda mendapatkan rekomendasi item terkait, dan ingin menerapkan beberapa aturan bisnis tertentu berdasarkan harga dan tanggal pembuatan yang bervariasi.

```
INCLUDE ItemID WHERE Items.PRICE < CurrentItem.PRICE AND Items.CREATION_TIMESTAMP >
 $DATE
```

Data pengguna

Ekspresi berikut mengecualikan item dengan genre atau genre yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan `$GENRE` parameter, tetapi hanya jika usia pengguna saat

ini sama dengan nilai yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan parameter. $AGE

```
EXCLUDE ItemID WHERE Items.GENRE IN ($GENRE) IF CurrentUser.AGE = $AGE
```

Ekspresi berikut hanya mencakup item dengan `watch` untuk CATEGORY_L1 dan `luxury` untuk CATEGORY_L2, jika usia pengguna saat ini sudah berakhir. 18

```
INCLUDE ItemID WHERE Items.CATEGORY_L1 IN ("watch") AND Items.CATEGORY_L2 IN ("luxury")
 IF CurrentUser.AGE > 18
```

Ekspresi filter segmen pengguna

Ekspresi filter berikut menunjukkan cara memfilter segmen pengguna berdasarkan data interaksi item dan metadata pengguna. Mereka diatur oleh tipe data.

Data pengguna

Ekspresi filter berikut hanya mencakup pengguna dengan status keanggotaan sama dengan nilai yang Anda tentukan saat Anda mendapatkan segmen pengguna.

```
INCLUDE UserID WHERE Users.MEMBERSHIP_STATUS IN ($MEMBERSHIP)
```

Ekspresi filter berikut mengecualikan pengguna dengan nilai AGE kurang dari yang Anda tentukan saat Anda mendapatkan segmen pengguna.

```
EXCLUDE UserID WHERE Users.AGE < $AGE
```

Data interaksi item

Ekspresi filter berikut hanya mencakup pengguna yang telah mengklik atau menilai item.

```
INCLUDE UserID WHERE Interactions.EVENT_TYPE IN ("click", "rating")
```

Ekspresi filter berikut mengecualikan pengguna dari segmen pengguna yang memiliki interaksi item dengan jenis peristiwa yang Anda tentukan saat Anda mendapatkan segmen pengguna.

```
EXCLUDE UserID WHERE Interactions.EVENT_TYPE IN ($EVENT_TYPE)
```

Contoh ekspresi filter rekomendasi tindakan

Contoh ekspresi filter berikut menunjukkan cara memfilter tindakan berdasarkan data interaksi tindakan, data tindakan. dan data pengguna. Mereka diatur oleh tipe data.

Topik

- Data interaksi aksi
- Data tindakan
- Data pengguna

Data interaksi aksi

Ekspresi filter berikut hanya mencakup tindakan dalam rekomendasi yang telah berinteraksi dengan pengguna, ketika interaksi tersebut memiliki jenis peristiwa yang Anda tentukan saat Anda mendapatkan rekomendasi.

```
INCLUDE ActionID WHERE Action_Interactions.EVENT_TYPE IN ($EVENT_TYPE)
```

Ekspresi filter berikut mengecualikan tindakan yang belum diambil pengguna berdasarkan jenis acara.

```
EXCLUDE ActionID WHERE Action_Interactions.EVENT_TYPE IN ("NOT_TAKEN")
```

Data tindakan

Ekspresi berikut mengecualikan tindakan berdasarkan kategori atau kategori yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan `$CATEGORY` parameter.

```
EXCLUDE ActionID WHERE Actions.CATEGORY IN ($CATEGORY)
```

Ekspresi berikut hanya mencakup tindakan dengan nilai lebih besar dari nilai yang Anda tentukan saat Anda mendapatkan rekomendasi.

```
INCLUDE ActionID WHERE Actions.VALUE > ($VALUE)
```

Data pengguna

Ekspresi berikut hanya mencakup tindakan untuk anggota premium jika pengguna saat ini memiliki keanggotaan premium.

```
INCLUDE ActionID WHERE Action.MEMBERSHIP_LEVEL IN ("Premium") IF CurrentUser.MEMBERSHIP
 = $PREMIUM
```

Ekspresi berikut mengecualikan tindakan dengan nilai VALUE kurang dari yang Anda tentukan saat Anda mendapatkan rekomendasi jika pengguna saat ini adalah anggota premium.

```
EXCLUDE ActionID WHERE Actions.VALUE < ($VALUE) IF CurrentUser.MEMBERSHIP = $PREMIUM
```

Menggabungkan beberapa ekspresi

Untuk menggabungkan beberapa ekspresi bersama-sama Anda menggunakan pemisah pipa (|). Gunakan kombinasi ekspresi saat Anda ingin menggunakan satu filter dan filter pada kumpulan data interaksi Item dan Item, atau kumpulan data interaksi Tindakan dan Tindakan. Setiap ekspresi pertama-tama dievaluasi secara independen dan hasilnya adalah penyatuan atau persimpangan dari dua hasil. Contoh berikut menunjukkan cara membuat ekspresi untuk kumpulan data interaksi Item dan Item, tetapi aturan yang sama berlaku saat bekerja dengan interaksi Tindakan dan Tindakan.

Contoh ekspresi yang cocok

Jika kedua ekspresi menggunakan EXCLUDE atau kedua ekspresi digunakanINCLUDE, hasilnya adalah penyatuan dari dua hasil sebagai berikut (A dan B adalah ekspresi yang berbeda):

- `Exclude A | Exclude B` adalah sama dengan `Exclude result from A or result from B`
- `Include A | Include B` adalah sama dengan `Include result from A or result from B`

Contoh berikut menunjukkan bagaimana menggabungkan dua ekspresi yang menggunakanINCLUDE. Ekspresi pertama hanya mencakup item dengan kategori atau kategori yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan $CATEGORY parameter. Ekspresi kedua mencakup item pengguna telah ditandai sebagai`favorite`. Rekomendasi hanya akan mencakup item dengan kategori yang Anda tentukan bersama dengan item yang telah ditandai pengguna sebagai favorit.

```
INCLUDE ItemID WHERE Items.CATEGORY IN ($CATEGORY) | INCLUDE ItemID WHERE
 Interactions.EVENT_TYPE IN ("favorite")
```

INCLUDE dan EXCLUDE contoh

Jika satu atau lebih ekspresi menggunakan `INCLUDE` dan satu lagi ekspresi menggunakan`EXCLUDE`, hasilnya adalah pengurangan hasil ekspresi dari hasil `EXCLUDE` ekspresi sebagai berikut (A, B, C, dan D adalah ekspresi yang berbeda). `INCLUDE`

- `Include A | Exclude B` adalah sama dengan `Include result from A - result from B`
- `Include A | Include B | Exclude C | Exclude D` adalah sama dengan `Include (A or B) - (C or D)`

Urutan ekspresi tidak masalah: Jika ekspresi EXCLUDE muncul sebelum ekspresi INCLUDE, hasilnya sama.

Contoh berikut menunjukkan bagaimana menggabungkan `INCLUDE` ekspresi dan `EXCLUDE` ekspresi. Ekspresi pertama hanya mencakup item dengan genre atau genre yang Anda tentukan saat Anda mendapatkan rekomendasi menggunakan `$GENRE` parameter. Ekspresi kedua mengecualikan item yang telah diklik atau dialirkan pengguna. Rekomendasi hanya akan mencakup item dengan genre yang Anda tentukan yang belum diklik atau dialirkan.

```
INCLUDE ItemID WHERE Items.GENRE IN ($GENRE) | EXCLUDE ItemID WHERE
 Interactions.EVENT_TYPE IN ("click", "stream")
```

Memfilter rekomendasi waktu nyata

Anda dapat memfilter rekomendasi real-time dengan konsol Amazon Personalize, AWS Command Line Interface (AWS CLI), atau. AWS SDKs

Ketika Anda mendapatkan rekomendasi item yang dipersonalisasi atau item serupa, Anda dapat menentukan promosi dalam permintaan Anda. Promosi menggunakan filter untuk menentukan aturan bisnis tambahan yang berlaku untuk subset item yang direkomendasikan yang dapat dikonfigurasi. Untuk informasi selengkapnya, lihat Mempromosikan item dalam rekomendasi waktu nyata.

Topik

- Memfilter rekomendasi waktu nyata (konsol)
- Memfilter rekomendasi waktu nyata ()AWS CLI
- Memfilter rekomendasi waktu nyata ()AWS SDKs

Memfilter rekomendasi waktu nyata (konsol)

Untuk memfilter rekomendasi real-time menggunakan konsol, buat filter dan kemudian terapkan ke permintaan rekomendasi.

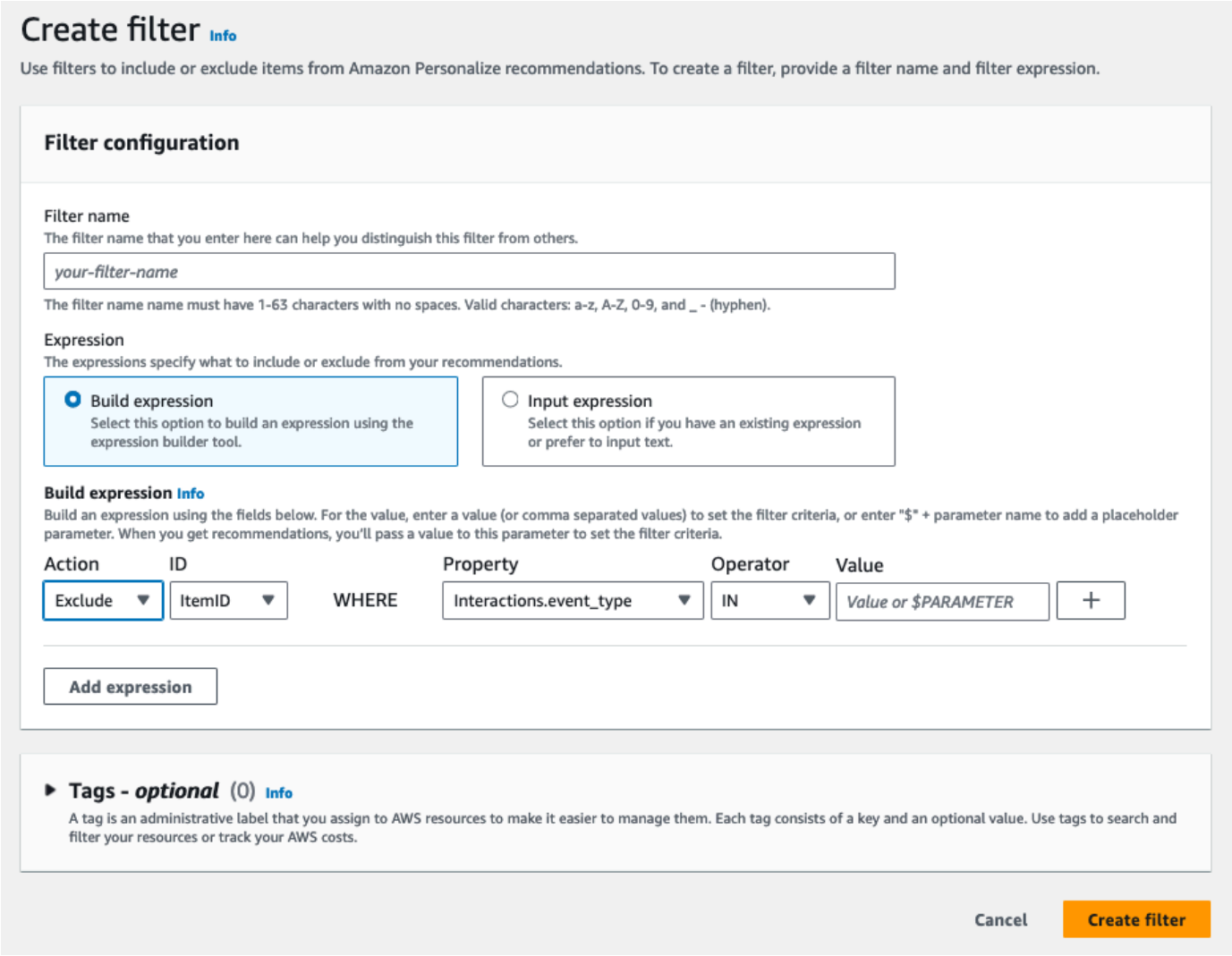
> **Note**
>
> Untuk memfilter rekomendasi menggunakan filter dengan parameter dan kampanye yang diterapkan sebelum 10 November 2020, Anda harus menerapkan ulang kampanye dengan menggunakan UpdateCampaign operasi atau membuat kampanye baru.

Membuat filter (konsol)

Untuk membuat filter di konsol, pilih grup kumpulan data yang berisi kampanye atau pemberi rekomendasi yang ingin Anda gunakan untuk mendapatkan rekomendasi yang difilter. Kemudian berikan nama filter dan ekspresi filter.

Untuk membuat filter (konsol)

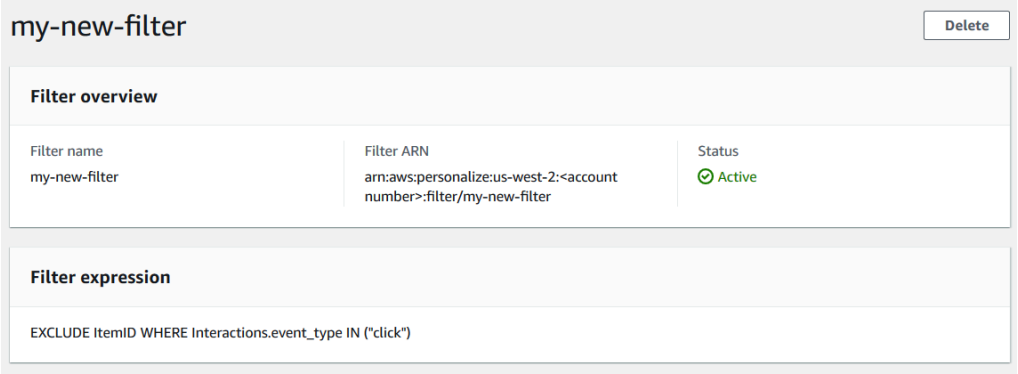
1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data yang berisi kampanye atau pemberi rekomendasi yang ingin Anda gunakan untuk mendapatkan rekomendasi yang difilter.
3. Di panel navigasi, pilih Filter dan kemudian pilih Buat filter baru. Halaman Create filter ditampilkan.



4. Untuk nama Filter, masukkan nama untuk filter Anda. Anda akan memilih filter dengan nama ini saat Anda menerapkannya pada permintaan rekomendasi.
5. Untuk Expression, pilih Build expression atau Add expression secara manual dan build atau masukkan ekspresi Anda:

   - Untuk menggunakan pembuat ekspresi, pilih Build expression. Pembuat ekspresi menyediakan struktur, bidang, dan pedoman untuk membuat ekspresi filter yang diformat dengan benar. Untuk informasi selengkapnya, lihat Menggunakan pembuat ekspresi filter.
   - Untuk memasukkan ekspresi Anda sendiri, pilih Tambahkan ekspresi secara manual. Untuk informasi selengkapnya, lihat Filter elemen ekspresi.
6. Pilih Selesai. Halaman ikhtisar filter menunjukkan nama sumber daya Amazon (ARN) filter, status, dan ekspresi filter lengkap. Untuk menghapus filter, pilih Hapus. Untuk informasi tentang

menemukan dan menghapus filter setelah Anda meninggalkan halaman ikhtisar, lihatMenghapus filter (konsol).



Menerapkan filter (konsol)

Untuk menerapkan filter, di Rekomendasi pengujian (untuk pemberi rekomendasi) atau Hasil kampanye uji (untuk kampanye khusus), pilih filter dan masukkan nilai parameter filter apa pun. Kemudian dapatkan rekomendasi untuk pengguna.

> **Important**
>
> Untuk ekspresi filter yang menggunakan `INCLUDE` elemen, Anda harus memberikan nilai untuk semua parameter yang didefinisikan dalam ekspresi. Untuk filter dengan ekspresi yang menggunakan `EXCLUDE` elemen, Anda dapat menghilangkan. `filter-values` Dalam kasus ini, Amazon Personalize tidak menggunakan bagian ekspresi tersebut untuk memfilter rekomendasi.

Untuk menerapkan filter (konsol)

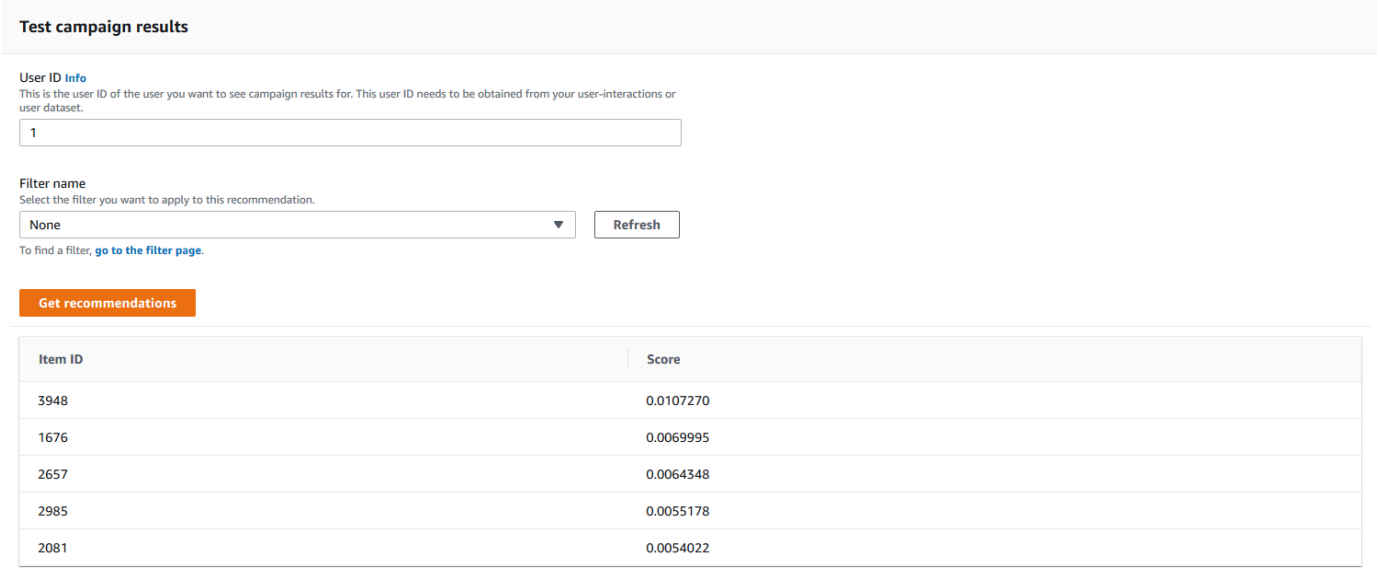
1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data yang berisi kampanye atau pemberi rekomendasi yang ingin Anda gunakan untuk mendapatkan rekomendasi yang difilter.
3. Bergantung pada jenis grup kumpulan data atau tipe sumber daya Anda, lakukan salah satu hal berikut:

   a. Untuk grup kumpulan data Domain, di panel navigasi pilih Rekomendasi.

b. Untuk grup kumpulan data kustom atau sumber daya kustom, di panel navigasi pilih Sumber daya kustom lalu Kampanye.

4. Pada halaman Rekomendasi atau Kampanye, pilih pemberi rekomendasi target atau kampanye.

5. Sebagai perbandingan, mulailah dengan mendapatkan rekomendasi tanpa menerapkan filter. Di bawah Rekomendasi pengujian/Hasil kampanye uji, masukkan ID pengguna yang ingin Anda rekomendasikan, atau ID item untuk item terkait, dan pilih Dapatkan rekomendasi. Tabel yang berisi rekomendasi teratas muncul.



6. Dari menu Filter name, pilih filter yang Anda buat. Jika filter Anda memiliki parameter placeholder, bidang terkait untuk setiap parameter akan muncul.

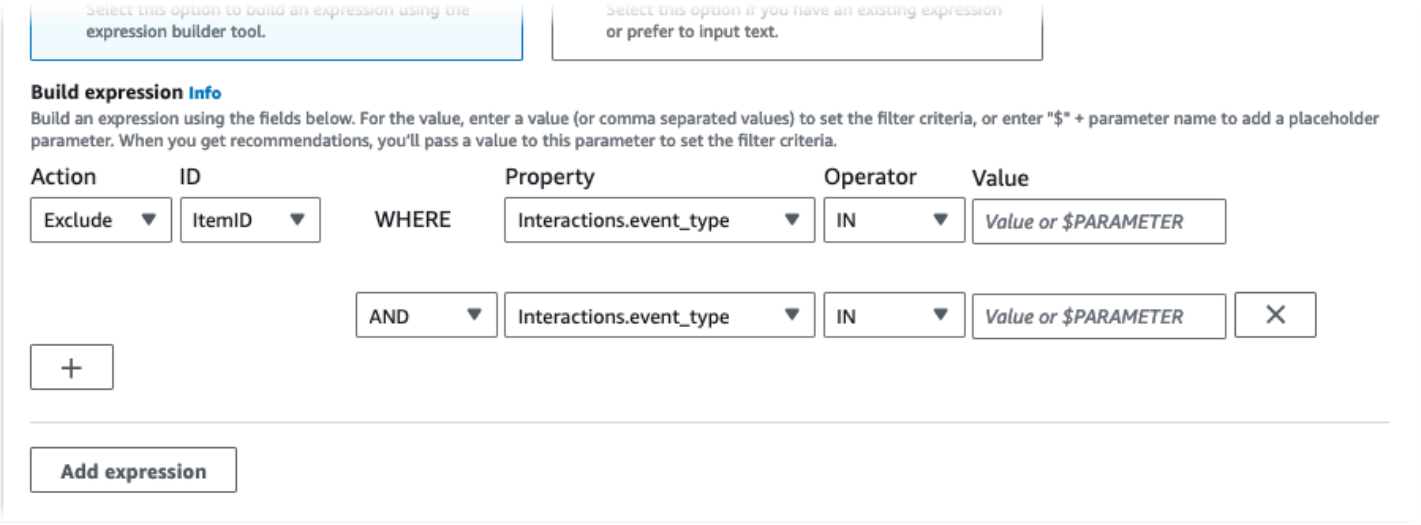
7. Jika Anda menggunakan filter dengan parameter placeholder, untuk setiap parameter, masukkan nilai untuk menetapkan kriteria filter. Untuk menggunakan beberapa nilai untuk satu parameter, pisahkan setiap nilai dengan koma.

8. Menggunakan yang sama `User ID` atau `Item ID` seperti pada langkah sebelumnya, pilih Dapatkan rekomendasi. Tabel rekomendasi muncul.

   Misalnya, jika pengguna sudah membeli item yang direkomendasikan, filter menghapusnya dari daftar rekomendasi. Dalam contoh ini, item 2657, 2985 digantikan oleh item yang paling cocok yang tidak dibeli pengguna (item 2641 dan 1573).

Menggunakan pembuat ekspresi filter

Pembuat Ekspresi pada halaman Buat filter menyediakan struktur, bidang, dan pedoman untuk membuat filter yang diformat dengan benar.



Untuk membangun ekspresi filter:

- Gunakan bidang Type, Action, Property, Operator, dan Value untuk membuat ekspresi.

  Untuk Nilai, masukkan nilai tetap atau, untuk menetapkan kriteria filter saat Anda mendapatkan rekomendasi, masukkan $ + nama parameter. Misalnya, `$GENRES`. Ketika Anda mendapatkan rekomendasi, Anda akan memberikan nilai atau nilai untuk difilter. Dalam contoh ini, Anda akan memberikan genre atau daftar genre ketika Anda mendapatkan rekomendasi.

  Pisahkan beberapa nilai non-parameter dengan koma. Anda tidak dapat menambahkan parameter yang dipisahkan koma ke filter.

  > **Note**
  >
  > Setelah Anda memilih Properti (dalam `dataset.field` format), nilai Properti untuk setiap baris berikutnya yang dirantai oleh `AND` atau `OR` kondisi harus menggunakan yang sama. `dataset`

- Gunakan tombol + dan X untuk menambah atau menghapus baris dari ekspresi Anda. Anda tidak dapat menghapus baris pertama.
- Untuk baris baru, gunakan `AND`IF,, atau `OR` operator pada menu AND untuk membuat rantai kondisi.

Untuk `IF` kondisi:

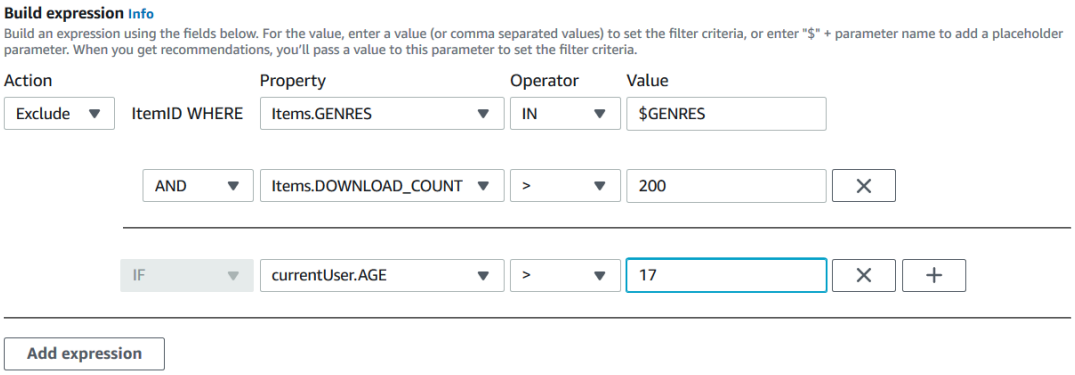
- Setiap ekspresi hanya dapat berisi satu `IF` item. Jika Anda menghapus kondisi IF, pembuat Ekspresi menghapus `AND` kondisi apa pun yang mengikutinya.
- Anda dapat menggunakan `IF` kondisi hanya untuk ekspresi yang memfilter oleh`CurrentUser`.

- Pilih Tambahkan ekspresi tombol untuk menambahkan ekspresi filter tambahan untuk pemfilteran yang lebih tepat. Setiap ekspresi pertama-tama dievaluasi secara independen dan hasilnya adalah penyatuan dari dua hasil.

> **Note**
>
> Untuk membuat filter yang menggunakan kumpulan data interaksi Item dan Item, atau kumpulan data interaksi Tindakan dan Tindakan, Anda harus menggunakan beberapa ekspresi.

Contoh pembuat ekspresi

Contoh berikut menunjukkan cara membuat filter yang mengecualikan item dengan genre yang Anda tentukan saat Anda mendapatkan rekomendasi (perhatikan parameter placeholder $GENRES). Filter juga mengecualikan item dengan lebih `DOWNLOAD_COUNT` dari`200`, tetapi hanya jika usia pengguna saat ini lebih besar dari`17`.



Menghapus filter (konsol)

Menghapus filter akan menghapus filter dari daftar filter untuk grup kumpulan data.

> **Important**
>
> Anda tidak dapat menghapus filter saat pekerjaan inferensi batch sedang berlangsung.

Untuk menghapus filter (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Dari daftar grup Dataset, pilih grup kumpulan data yang berisi filter yang ingin Anda hapus.
3. Di panel navigasi, pilih Filter.
4. Dari daftar filter, pilih filter yang ingin Anda hapus dan pilih Lihat Detail. Halaman detail filter muncul.
5. Pilih Hapus dan konfirmasikan penghapusan di kotak dialog konfirmasi.

Memfilter rekomendasi waktu nyata ()AWS CLI

Untuk memfilter rekomendasi menggunakan AWS CLI, Anda membuat filter dan kemudian menerapkannya dengan menentukan filter ARN dalam permintaan GetRecommendations atauGetPersonalizedRanking.

> **Important**
>
> Untuk memfilter rekomendasi menggunakan filter dengan parameter dan kampanye yang Anda gunakan sebelum 10 November 2020, Anda harus menerapkan ulang kampanye dengan menggunakan UpdateCampaign panggilan atau membuat kampanye baru.

Membuat filter (AWS CLI)

Gunakan `create-filter` operasi berikut untuk membuat filter dan menentukan ekspresi filter.

Ganti `Filter name` dengan nama filter, dan `Dataset group ARN` dengan Amazon Resource Name (ARN) dari grup dataset. Ganti sampel `filter-expression` dengan ekspresi filter Anda sendiri.

```
aws personalize create-filter \
  --name Filter name \
```

```
  --dataset-group-arn dataset group arn \
  --filter-expression "EXCLUDE ItemID WHERE Items.CATEGORY IN (\"$CATEGORY\")"
```

Jika berhasil, filter ARN ditampilkan. Rekam untuk digunakan nanti. Untuk memverifikasi bahwa filter aktif, gunakan DescribeFilter operasi sebelum Anda menggunakan filter.

Untuk informasi selengkapnya tentang API, lihat CreateFilter. Untuk informasi selengkapnya tentang ekspresi filter, termasuk contoh, lihatFilter struktur ekspresi dan elemen.

Menerapkan filter (AWS CLI)

Saat Anda menggunakan`get-recommendations`, `get-action-recommendations` atau `get-personalized-ranking` operasi, Anda menerapkan filter dengan meneruskan `filter-arn` dan nilai filter apa pun sebagai parameter.

Berikut ini adalah contoh `get-recommendations` operasi. Ganti `Campaign ARN` dengan Nama Sumber Daya Amazon (ARN) kampanye Anda `User ID` dengan ID pengguna yang Anda dapatkan rekomendasinya, dan `Filter ARN` dengan ARN filter Anda. Jika Anda mendapatkan rekomendasi dari pemberi rekomendasi alih-alih kampanye, gunakan `recommender-arn` alih-alih `--campaign-arn` dan berikan ARN untuk pemberi rekomendasi.

Jika ekspresi Anda memiliki parameter apa pun, sertakan `filter-values` objek. Untuk setiap parameter dalam ekspresi filter Anda, berikan nama parameter (case sensitive) dan nilainya. Misalnya, jika ekspresi filter Anda memiliki `$GENRE` parameter, berikan “GENRE” sebagai kunci, dan genre atau genre, seperti`"Comedy"`, sebagai nilai. Pisahkan beberapa nilai dengan koma. Misalnya, `"\"comedy\",\"drama\",\"horror\""`.

> **Important**
>
> Untuk ekspresi filter yang menggunakan `INCLUDE` elemen untuk menyertakan item, Anda harus memberikan nilai untuk semua parameter yang ditentukan dalam ekspresi. Untuk filter dengan ekspresi yang menggunakan `EXCLUDE` elemen untuk mengecualikan item, Anda dapat menghilangkan. `filter-values` Dalam kasus ini, Amazon Personalize tidak menggunakan bagian ekspresi tersebut untuk memfilter rekomendasi.

```
aws personalize-runtime get-recommendations \
  --campaign-arn Campaign ARN \
  --user-id User ID \
  --filter-arn Filter ARN \
```

```
  --filter-values '{
      "Parameter name": "\"value\"",
      "Parameter name": "\"value1\",\"value2\",\"value3\""
    }'
```

Menghapus filter ()AWS CLI

Gunakan `delete-filter` operasi berikut untuk menghapus filter. Ganti `filter ARN` dengan ARN filter.

```
aws personalize delete-filter --filter-arn Filter ARN
```

Memfilter rekomendasi waktu nyata ()AWS SDKs

Untuk memfilter rekomendasi menggunakan AWS SDKs, Anda membuat filter dan kemudian menerapkannya dengan menentukan filter ARN dalam permintaan GetRecommendations atauGetPersonalizedRanking.

> **Important**
>
> Untuk memfilter rekomendasi menggunakan filter dengan parameter dan kampanye yang Anda gunakan sebelum 10 November 2020, Anda harus menerapkan ulang kampanye dengan menggunakan UpdateCampaign panggilan atau membuat kampanye baru.

Membuat filter (AWS SDKs)

Buat filter baru dengan CreateFilter operasi. Kode berikut menunjukkan cara membuat filter. Tentukan nama filter, Nama Sumber Daya Amazon (ARN) grup kumpulan data Anda, dan berikan ekspresi filter Anda.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.create_filter(
    name = 'Filter Name',
    datasetGroupArn = 'Dataset Group ARN',
```

```
    filterExpression = 'EXCLUDE ItemID WHERE Items.CATEGORY IN ($CATEGORY)'
)
filter_arn = response["filterArn"]
print("Filter ARN: " + filter_arn)
```

SDK for Java 2.x

```
public static String createFilter(PersonalizeClient personalizeClient,
                                  String filterName,
                                  String datasetGroupArn,
                                  String filterExpression) {
    try {
        CreateFilterRequest request = CreateFilterRequest.builder()
                .name(filterName)
                .datasetGroupArn(datasetGroupArn)
                .filterExpression(filterExpression)
                .build();

        return personalizeClient.createFilter(request).filterArn();
    }
    catch(PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateFilterCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the filter's parameters.
export const createFilterParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  name: "NAME" /* required */,
  filterExpression: "FILTER_EXPRESSION" /*required */,
};

export const run = async () => {
```

```
  try {
    const response = await personalizeClient.send(
      new CreateFilterCommand(createFilterParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Rekam filter ARN untuk digunakan nanti. Untuk memverifikasi bahwa filter aktif, gunakan DescribeFilter operasi sebelum menggunakan filter. Untuk informasi selengkapnya tentang API, lihat CreateFilter. Untuk informasi selengkapnya tentang ekspresi filter, termasuk contoh, lihatFilter struktur ekspresi dan elemen.

Menerapkan filter (AWS SDKs)

Saat Anda menggunakan GetRecommendations, GetActionRecommendations, atau GetPersonalizedRanking operasi, terapkan filter dengan melewatkan nilai filter `filterArn` dan apa pun sebagai parameter.

Kode berikut menunjukkan cara mendapatkan rekomendasi item Amazon Personalize yang difilter untuk pengguna. Tentukan ID pengguna yang ingin Anda rekomendasikan, Nama Sumber Daya Amazon (ARN) kampanye Anda, dan ARN filter Anda. Jika Anda mendapatkan rekomendasi dari pemberi rekomendasi alih-alih kampanye, gunakan `recommenderArn` alih-alih `campaignArn` dan berikan ARN untuk pemberi rekomendasi.

Untuk`filterValues`, untuk setiap parameter opsional dalam ekspresi filter Anda, berikan nama parameter (peka huruf besar/kecil) dan nilai atau nilainya. Misalnya, jika ekspresi filter Anda memiliki `$GENRES` parameter, berikan “GENRES” sebagai kunci, dan genre atau genre, seperti`"\"Comedy"\"`, sebagai nilai. Untuk beberapa nilai, pisahkan setiap nilai dengan koma. Misalnya, `"\"comedy\",\"drama\",\"horror\""`.

> **Important**
>
> Untuk ekspresi filter yang menggunakan `INCLUDE` elemen untuk menyertakan item, Anda harus memberikan nilai untuk semua parameter yang ditentukan dalam ekspresi. Untuk filter dengan ekspresi yang menggunakan `EXCLUDE` elemen untuk mengecualikan item,

Anda dapat menghilangkan. `filter-values` Dalam kasus ini, Amazon Personalize tidak menggunakan bagian ekspresi tersebut untuk memfilter rekomendasi.

SDK for Python (Boto3)

```
import boto3

personalize_runtime = boto3.client("personalize-runtime")

response = personalize_runtime.get_recommendations(
    campaignArn = "Campaign ARN",
    userId = "User ID",
    filterArn = "Filter ARN",
    filterValues = {
      "Parameter name": "\"value1\"",
      "Parameter name": "\"value1\",\"value2\",\"value3\""
      ....
    }
)
```

SDK for Java 2.x

Contoh berikut menggunakan dua parameter, satu dengan dua nilai dan satu dengan satu nilai. Bergantung pada ekspresi filter Anda, ubah kode untuk menambah atau menghapus bidang ParameterName dan ParameterValue.

```
public static void getFilteredRecs(PersonalizeRuntimeClient
 personalizeRuntimeClient,
                                    String campaignArn,
                                    String userId,
                                    String filterArn,
                                    String parameter1Name,
                                    String parameter1Value1,
                                    String parameter1Value2,
                                    String parameter2Name,
                                    String parameter2Value){

    try {

        Map<String, String> filterValues = new HashMap<>();
```

```
        filterValues.put(parameter1Name, String.format("\"%1$s\",\"%2$s\"",
                parameter1Value1, parameter1Value2));
        filterValues.put(parameter2Name, String.format("\"%1$s\"",
                parameter2Value));

        GetRecommendationsRequest recommendationsRequest =
 GetRecommendationsRequest.builder()
                .campaignArn(campaignArn)
                .numResults(20)
                .userId(userId)
                .filterArn(filterArn)
                .filterValues(filterValues)
                .build();

        GetRecommendationsResponse recommendationsResponse =
 personalizeRuntimeClient.getRecommendations(recommendationsRequest);
        List<PredictedItem> items = recommendationsResponse.itemList();

        for (PredictedItem item: items) {
            System.out.println("Item Id is : "+item.itemId());
            System.out.println("Item score is : "+item.score());
        }
    } catch (PersonalizeRuntimeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { GetRecommendationsCommand } from "@aws-sdk/client-personalize-runtime";
import { personalizeRuntimeClient } from "./libs/personalizeClients.js";
// Or, create the client here:
// const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region:
 "REGION"});

// Set recommendation request parameters.
export const getRecommendationsParam = {
  campaignArn: "CAMPAIGN_ARN" /* required */,
  userId: "USER_ID" /* required */,
  numResults: 15 /* optional */,
  filterArn: "FILTER_ARN" /* required to filter recommendations */,
```

```
  filterValues: {
    PROPERTY:
      '"VALUE"' /* Only required if your filter has a placeholder parameter */,
  },
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(
      new GetRecommendationsCommand(getRecommendationsParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Menghapus filter (AWS Python SDK)

Gunakan `delete_filter` metode berikut untuk menghapus filter. Ganti `filter ARN` dengan ARN filter.

```
import boto3
personalize = boto3.client("personalize")

response = personalize.delete_filter(
  filterArn = "filter ARN"
)
```

Memfilter rekomendasi batch dan segmen pengguna (sumber daya khusus)

Memfilter rekomendasi batch dan segmen pengguna bekerja hampir sama dengan memfilter rekomendasi real-time. Ini mengikuti alur kerja yang sama yang dijelaskan dalam Mendapatkan rekomendasi item batch atauMendapatkan segmen pengguna batch. Untuk memfilter rekomendasi batch atau segmen pengguna, Anda melakukan hal berikut:

1. Buat filter seperti yang Anda lakukan untuk rekomendasi waktu nyata. Untuk informasi selengkapnya, lihat Memfilter rekomendasi waktu nyata.
2. Siapkan data input Anda dan unggah ke Amazon S3 seperti yang dijelaskan dalam Mempersiapkan data input untuk rekomendasi batch atau. Mempersiapkan data input untuk segmen pengguna Jika filter Anda menggunakan parameter placeholder, Anda harus menambahkan objek tambahan`filterValues`. Untuk informasi selengkapnya, lihat Memberikan nilai filter di JSON masukan Anda. Jika filter Anda tidak menggunakan parameter placeholder, data input Anda dapat mengikuti contoh di Contoh input dan keluaran pekerjaan inferensi Batch JSON Contoh JSON masukan dan keluaran pekerjaan segmen Batch
3. Buat lokasi terpisah untuk data keluaran Anda, baik folder atau bucket Amazon S3 lainnya.
4. Buat pekerjaan inferensi batch atau pekerjaan segmen batch. Saat Anda membuat pekerjaan, tentukan Nama Sumber Daya Amazon (ARN) filter Anda.
5. Saat inferensi batch atau tugas segmen batch selesai, ambil rekomendasi atau segmen pengguna dari lokasi keluaran Anda di Amazon S3.

Topik

- Memberikan nilai filter di JSON masukan Anda
- Memfilter alur kerja batch (konsol)
- Memfilter alur kerja batch ()AWS SDKs

Memberikan nilai filter di JSON masukan Anda

Untuk filter dengan parameter placeholder, seperti`$GENRE`, Anda harus memberikan nilai untuk parameter dalam `filterValues` objek di JSON input Anda. Untuk sebuah `filterValues` objek, setiap kunci adalah nama parameter. Setiap nilai adalah kriteria yang Anda lewati sebagai parameter. Kelilingi setiap nilai dengan tanda kutip yang lolos:. `"filterValues": {"GENRES":"\"drama\""}` Untuk beberapa nilai, pisahkan setiap nilai dengan koma: `"filterValues":{"GENRES":"\"horror\",\"comedy\",\"drama\""}`

Contoh input pekerjaan inferensi Batch JSON

Berikut ini adalah contoh dari beberapa baris pertama dari file input JSON untuk pekerjaan inferensi batch. Contohnya termasuk `filterValues` objek. `GENRES`Kuncinya sesuai dengan `$GENRES` placeholder dalam ekspresi filter. Pekerjaan dalam contoh ini menggunakan resep User-

Personalization. Untuk resep RELATED_ITEMS, berikan ItemId alih-alih userId. Untuk resep PERSONALIZED_RANKING berikan userId dan ItemList.

```
{"userId": "5","filterValues":{"GENRES":"\"horror\",\"comedy\",\"drama\""}}
{"userId": "3","filterValues":{"GENRES":"\"horror\",\"comedy\""}}
{"userId": "34","filterValues":{"GENRES":"\"drama\""}}
```

Untuk contoh lebih lanjut dari data input pekerjaan inferensi batch berdasarkan resep, lihatContoh input dan keluaran pekerjaan inferensi Batch JSON. Anda dapat menggunakan contoh-contoh ini sebagai titik awal dan menambahkan `filterValues` objek dari contoh di atas.

Masukan pekerjaan segmen Batch Contoh JSON

Berikut ini adalah contoh dari beberapa baris pertama dari file input JSON dengan nilai filter untuk pekerjaan segmen batch. `GENRES`Kuncinya sesuai dengan `$GENRES` placeholder dalam ekspresi filter.

```
{"itemAttributes": "ITEMS.genres = \"Comedy\" AND ITEMS.genres = \"Action
\"","filterValues":{"COUNTRY":"\"Japan\""}}
{"itemAttributes": "ITEMS.genres = \"Horror\"","filterValues":{"COUNTRY":"\"United
 States\"\""}}
{"itemAttributes": "ITEMS.genres = \"Action\" AND ITEMS.genres = \"Adventure
\"","filterValues":{"COUNTRY":"\"England\""}}
```

Untuk contoh lebih lanjut dari data input pekerjaan inferensi batch berdasarkan resep, lihatContoh JSON masukan dan keluaran pekerjaan segmen Batch. Anda dapat menggunakan contoh-contoh ini sebagai titik awal dan menambahkan `filterValues` objek dari contoh di atas.

Memfilter alur kerja batch (konsol)

Untuk memfilter alur kerja batch dengan konsol Amazon Personalize, Anda membuat filter dan kemudian Anda membuat pekerjaan inferensi batch atau tugas segmen batch dan memilih filter. Untuk petunjuk langkah demi langkah lengkap, lihat Membuat pekerjaan inferensi batch (konsol) danMembuat pekerjaan segmen batch (konsol).

Memfilter alur kerja batch ()AWS SDKs

Untuk memfilter rekomendasi batch dengan AWS SDKs, buat filter dan sertakan `FilterArn` parameter dalam CreateBatchSegmentJob permintaan CreateBatchInferenceJob atau.

Kode berikut menunjukkan cara membuat pekerjaan inferensi batch dengan filter menggunakan. AWS SDK untuk Python (Boto3) Sebaiknya gunakan lokasi yang berbeda untuk data keluaran Anda (baik folder atau bucket Amazon S3 yang berbeda). Untuk penjelasan lengkap tentang semua bidang, lihatMembuat pekerjaan inferensi batch ()AWS SDKs.

```
import boto3

personalize = boto3.client("personalize")

personalize_rec.create_batch_inference_job (
    solutionVersionArn = "Solution version ARN",
    jobName = "Batch job name",
    roleArn = "IAM role ARN",
    filterArn = "Filter ARN",
    jobInput =
        {"s3DataSource": {"path": "S3 input path"}},
    jobOutput =
        {"S3DataDestination": {"path": "S3 output path"}}
)
```

Merekam peristiwa waktu nyata untuk memengaruhi rekomendasi

Peristiwa adalah interaksi antara pengguna dan katalog Anda. Ini bisa berupa interaksi dengan item, seperti pengguna yang membeli item atau menonton video, atau dapat mengambil tindakan, seperti mengajukan kartu kredit atau mendaftar dalam program keanggotaan.

Amazon Personalize dapat membuat rekomendasi berdasarkan data peristiwa waktu nyata saja, data peristiwa historis saja, atau campuran keduanya. Rekam peristiwa real-time saat pelanggan Anda berinteraksi dengan rekomendasi. Ini membangun data interaksi Anda dan membuat data Anda tetap segar. Dan itu memberi tahu Amazon Personalisasi tentang minat pengguna Anda saat ini, yang dapat meningkatkan relevansi rekomendasi.

Jika kasus penggunaan domain atau resep khusus Anda mendukung personalisasi waktu nyata, Amazon Personalize menggunakan peristiwa secara real time untuk memperbarui dan menyesuaikan rekomendasi sesuai dengan minat pengguna yang terus berkembang.

Cara Anda merekam peristiwa real-time tergantung pada jenis data interaksi yang Anda impor:

- Untuk interaksi item, Anda merekam peristiwa real-time dengan operasi PutEvents API. Amazon Personalize menambahkan data ini ke data interaksi item di grup kumpulan data Anda. Untuk informasi selengkapnya, lihat Merekam peristiwa interaksi item waktu nyata.
- Untuk interaksi tindakan, Anda merekam peristiwa real-time dengan operasi PutActionInteractions API. Amazon Personalize menambahkan data ini ke kumpulan data interaksi Tindakan di grup kumpulan data Anda. Hanya resep PERSONALIZED_ACTIONS yang menggunakan data interaksi tindakan. Untuk informasi selengkapnya, lihat Merekam peristiwa interaksi aksi waktu nyata.

Topik

- Bagaimana peristiwa waktu nyata memengaruhi rekomendasi
- Merekam peristiwa interaksi item waktu nyata
- Merekam peristiwa interaksi aksi waktu nyata
- Merekam acara untuk pengguna anonim
- Layanan pelacakan acara pihak ketiga
- Contoh implementasi

Bagaimana peristiwa waktu nyata memengaruhi rekomendasi

Jika resep Anda mendukung personalisasi waktu nyata, setelah Anda membuat pemberi rekomendasi atau kampanye khusus, Amazon Personalize menggunakan data peristiwa baru yang direkam untuk item atau tindakan yang ada dalam hitungan detik setelah impor. Kasus penggunaan dan resep berikut mendukung personalisasi waktu nyata:

- Direkomendasikan untuk Anda (kasus penggunaan ECOMMERCE)
- Pilihan teratas untuk Anda (kasus penggunaan VIDEO_ON_DEMAND)
- Resep personalisasi-v2 pengguna
- Resep Personalisasi Pengguna
- Resep peringkat-v2 yang dipersonalisasi
- Resep Peringkat yang Dipersonalisasi
- Next-Best-Action resep

Jika Anda menggunakan resep Trending-Now, Amazon Personalize secara otomatis mempertimbangkan item dari data peristiwa baru melalui interval yang dapat dikonfigurasi. Anda tidak perlu membuat versi solusi baru. Untuk informasi selengkapnya, lihat Resep Trend-Now.

Jika item, tindakan, atau pengguna dalam acara tersebut baru, cara Amazon Personalisasi menggunakan data tergantung pada kasus penggunaan atau resep Anda. Untuk informasi selengkapnya, lihat Memperbarui data dalam kumpulan data setelah pelatihan.

Merekam peristiwa interaksi item waktu nyata

Peristiwa interaksi item adalah interaksi antara pengguna dan item dalam katalog Anda. Misalnya, pengguna membeli sepatu atau menonton film.

Rekam peristiwa interaksi item waktu nyata saat Anda menunjukkan rekomendasi item pelanggan Anda. Ini membangun data interaksi Anda dan membuat data Anda tetap segar. Dan itu memberi tahu Amazon Personalisasi tentang minat pengguna Anda saat ini, yang dapat meningkatkan relevansi rekomendasi.

Anda merekam peristiwa interaksi item dengan operasi PutEvents API. Amazon Personalize menambahkan data peristiwa ke kumpulan data interaksi Item di grup kumpulan data Anda. Jika

Anda merekam dua peristiwa dengan stempel waktu yang sama persis dan properti yang identik, Amazon Personalize hanya menyimpan salah satu acara. Anda dapat merekam peristiwa interaksi item menggunakan AWS SDKs, AWS Amplify, atau AWS Command Line Interface ()AWS CLI.

Jika Anda menggunakan Apache Kafka, Anda dapat menggunakan konektor Kafka untuk Amazon Personalize untuk mengalirkan interaksi item secara real time ke Amazon Personalize. Untuk informasi, lihat Konektor Kafka untuk Amazon Personalisasi di repositori Github personalize-kafka-connector.

AWS Amplify menyertakan JavaScript pustaka untuk merekam peristiwa interaksi item dari aplikasi klien web, dan pustaka untuk merekam peristiwa dalam kode server. Untuk informasi selengkapnya, lihat Amplify documentation.

Topik

- Persyaratan untuk merekam acara interaksi item dan melatih model
- Membuat pelacak peristiwa interaksi item
- Merekam peristiwa interaksi item tunggal
- Merekam beberapa peristiwa interaksi item dengan data nilai acara
- Merekam peristiwa interaksi item dengan data tayangan
- Metrik peristiwa dan laporan atribusi

Persyaratan untuk merekam acara interaksi item dan melatih model

Untuk merekam peristiwa interaksi item, Anda memerlukan yang berikut ini:

- Grup dataset yang menyertakan kumpulan `Item interactions` data, yang bisa kosong. Jika Anda membaca Memulai tutorial panduan ini, Anda dapat menggunakan grup kumpulan data dan kumpulan data yang sama dengan yang Anda buat. Untuk informasi tentang membuat grup kumpulan data dan kumpulan data, lihat. Mengimpor data pelatihan ke Amazon Personalize dataset
- Pelacak acara.
- Panggilan ke operasi PutEvents API.
- Jika Anda menggunakan AWS Lambda fungsi untuk memanggil PutEvents operasi, peran eksekusi fungsi Anda harus memiliki izin untuk melakukan `personalize:PutEvents` tindakan dengan wildcard `*` dalam `Resource` elemen.

Anda dapat memulai dengan kumpulan data interaksi Item kosong dan, ketika Anda telah merekam cukup data, latih model hanya menggunakan peristiwa baru yang direkam. Untuk semua kasus penggunaan (Grup kumpulan data domain) dan resep (Grup kumpulan data khusus), data interaksi Anda harus memiliki yang berikut sebelum pelatihan:

- Minimal 1000 catatan interaksi item dari pengguna yang berinteraksi dengan item di katalog Anda. Interaksi ini dapat berasal dari impor massal, atau acara streaming, atau keduanya.
- Minimal 25 pengguna unik IDs dengan setidaknya dua interaksi item untuk masing-masing.

Untuk rekomendasi kualitas, kami sarankan Anda memiliki minimal 50.000 interaksi item dari setidaknya 1.000 pengguna dengan dua atau lebih interaksi item masing-masing.

Membuat pelacak peristiwa interaksi item

Sebelum Anda dapat merekam peristiwa interaksi item, Anda harus membuat pelacak peristiwa interaksi item. Pelacak peristiwa mengarahkan data peristiwa baru ke kumpulan data interaksi Item di grup kumpulan data Anda.

Anda membuat pelacak peristiwa dengan konsol Amazon Personalize atau operasi APICreateEventTracker. Anda meneruskan sebagai parameter Nama Sumber Daya Amazon (ARN) dari grup kumpulan data yang berisi kumpulan data interaksi Item target. Untuk petunjuk cara membuat pelacak peristiwa menggunakan konsol Amazon Personalize, lihat. Membuat pelacak acara (konsol)

Pelacak peristiwa menyertakan ID pelacakan, yang Anda lewati sebagai parameter saat Anda menggunakan PutEventsoperasi. Amazon Personalize kemudian menambahkan data peristiwa baru ke kumpulan data interaksi Item dari grup kumpulan data yang Anda tentukan di pelacak peristiwa.

> **Note**
>
> Anda hanya dapat membuat satu pelacak peristiwa interaksi item untuk grup kumpulan data.

Python

```
import boto3

personalize = boto3.client('personalize')
```

```
response = personalize.create_event_tracker(
    name='MovieClickTracker',
    datasetGroupArn='arn:aws:personalize:us-west-2:acct-id:dataset-group/
MovieClickGroup'
)
print(response['eventTrackerArn'])
print(response['trackingId'])
```

Pelacak acara ARN dan tracking ID menampilkan, misalnya:

```
{
    "eventTrackerArn": "arn:aws:personalize:us-west-2:acct-id:event-tracker/
MovieClickTracker",
    "trackingId": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
}
```

AWS CLI

```
aws personalize create-event-tracker \
    --name MovieClickTracker \
    --dataset-group-arn arn:aws:personalize:us-west-2:acct-id:dataset-group/
MovieClickGroup
```

Pelacak acara ARN dan tracking ID menampilkan, misalnya:

```
{
    "eventTrackerArn": "arn:aws:personalize:us-west-2:acct-id:event-tracker/
MovieClickTracker",
    "trackingId": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
}
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { CreateEventTrackerCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the event tracker's parameters.
export const createEventTrackerParam = {
```

```
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  name: "NAME" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateEventTrackerCommand(createEventTrackerParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

SDK for Java 2.x

```
public static String createEventTracker(PersonalizeClient personalizeClient,
                                        String eventTrackerName,
                                        String datasetGroupArn) {

    String eventTrackerId = null;
    String eventTrackerArn = null;
    long maxTime = 3 * 60 * 60;
    long waitInMilliseconds = 30 * 1000;
    String status;

    try {
        CreateEventTrackerRequest createEventTrackerRequest =
 CreateEventTrackerRequest.builder()
            .name(eventTrackerName)
            .datasetGroupArn(datasetGroupArn)
            .build();

        CreateEventTrackerResponse createEventTrackerResponse =
            personalizeClient.createEventTracker(createEventTrackerRequest);

        eventTrackerArn = createEventTrackerResponse.eventTrackerArn();
        eventTrackerId = createEventTrackerResponse.trackingId();

        System.out.println("Event tracker ARN: " + eventTrackerArn);
```

```
        System.out.println("Event tracker ID: " + eventTrackerId);

        maxTime = Instant.now().getEpochSecond() + maxTime;

        DescribeEventTrackerRequest describeRequest =
 DescribeEventTrackerRequest.builder()
            .eventTrackerArn(eventTrackerArn)
            .build();

        while (Instant.now().getEpochSecond() < maxTime) {

            status =
 personalizeClient.describeEventTracker(describeRequest).eventTracker().status();
            System.out.println("EventTracker status: " + status);

            if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
                break;
            }
            try {
                Thread.sleep(waitInMilliseconds);
            } catch (InterruptedException e) {
                System.out.println(e.getMessage());
            }
        }
        return eventTrackerId;
    }
    catch (PersonalizeException e){
        System.out.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return eventTrackerId;
}
```

Merekam peristiwa interaksi item tunggal

Setelah Anda membuat kumpulan data interaksi Item dan pelacak peristiwa untuk grup kumpulan data Anda, Anda siap merekam peristiwa interaksi item. Contoh berikut menunjukkan `PutEvents` operasi yang melewati satu peristiwa interaksi item. Skema yang sesuai ditampilkan, bersama dengan baris contoh dari kumpulan data interaksi Item.

Aplikasi Anda menghasilkan unik `sessionId` ketika pengguna pertama kali mengunjungi situs web Anda atau menggunakan aplikasi Anda. Anda harus menggunakan hal yang sama

`sessionId` di semua acara sepanjang sesi. Amazon Personalize menggunakan `sessionId` untuk mengaitkan peristiwa dengan pengguna sebelum mereka masuk (bersifat anonim). Untuk informasi selengkapnya, lihat Merekam acara untuk pengguna anonim.

Daftar acara adalah array Event objek. An `eventType` diperlukan untuk setiap acara. Jika Anda tidak memiliki data tipe acara, Anda dapat memberikan nilai placeholder untuk memenuhi persyaratan.

Itu `trackingId` berasal dari pelacak acara yang Anda buat. Membuat pelacak peristiwa interaksi item `sentAt`Parameter`userId`,`itemId`, dan dipetakan ke bidang USER_ID, ITEM_ID, dan TIMESTAMP dari kumpulan data historis yang sesuai. `Interactions` Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

Kolom dataset yang sesuai

```
Dataset columns: USER_ID, ITEM_ID, TIMESTAMP, EVENT_TYPE
Example data: user123, item-xyz, 1543631760, click
```

Contoh kode

SDK for Python (Boto3)

```
import boto3

personalize_events = boto3.client(service_name='personalize-events')

personalize_events.put_events(
    trackingId = 'tracking_id',
    userId= 'USER_ID',
    sessionId = 'session_id',
    eventList = [{
        'sentAt': 1719511760,
        'eventType': 'click',
        'itemId': 'ITEM_ID'
        }]
)
```

SDK for JavaScript v3

```
// Get service clients module and commands using ES6 syntax.
import { PutEventsCommand } from "@aws-sdk/client-personalize-events";
import { personalizeEventsClient } from "./libs/personalizeClients.js";
```

```
// Or, create the client here.
// const personalizeEventsClient = new PersonalizeEventsClient({ region: "REGION"});

// Convert your UNIX timestamp to a Date.
const sentAtDate = new Date(1613443801 * 1000); // 1613443801 is a testing value.
 Replace it with your sentAt timestamp in UNIX format.

// Set put events parameters.
const putEventsParam = {
  eventList: [
    /* required */
    {
      eventType: "EVENT_TYPE" /* required */,
      sentAt: sentAtDate /* required, must be a Date with js */,
      eventId: "EVENT_ID" /* optional */,
      itemId: "ITEM_ID" /* optional */,
    },
  ],
  sessionId: "SESSION_ID" /* required */,
  trackingId: "TRACKING_ID" /* required */,
  userId: "USER_ID" /* required */,
};
export const run = async () => {
  try {
    const response = await personalizeEventsClient.send(
      new PutEventsCommand(putEventsParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

AWS CLI

```
aws personalize-events put-events \
    --tracking-id tracking_id \
    --user-id USER_ID \
    --session-id session_id \
    --event-list '[{
        "sentAt": 1719511760,
```

```
        "eventType": "click",
        "itemId": "ITEM_ID"
      }]'
```

SDK for Java 2.x

```
public static void putEvents(PersonalizeEventsClient personalizeEventsClient,
                             String trackingId,
                             String sessionId,
                             String userId,
                             String itemId,
                             String eventType) {

    try {
        Event event = Event.builder()
            .sentAt(Instant.ofEpochMilli(System.currentTimeMillis() + 10 * 60 *
 1000))
            .itemId(itemId)
            .eventType(eventType)
            .build();

        PutEventsRequest putEventsRequest = PutEventsRequest.builder()
            .trackingId(trackingId)
            .userId(userId)
            .sessionId(sessionId)
            .eventList(event)
            .build();

        int responseCode = personalizeEventsClient.putEvents(putEventsRequest)
            .sdkHttpResponse()
            .statusCode();
        System.out.println("Response code: " + responseCode);

        } catch (PersonalizeEventsException e) {
            System.out.println(e.awsErrorDetails().errorMessage());
        }
}
```

Merekam beberapa peristiwa interaksi item dengan data nilai acara

Setelah Anda membuat kumpulan data interaksi Item dan pelacak peristiwa untuk grup kumpulan data Anda, Anda siap merekam peristiwa interaksi item. Contoh berikut menunjukkan cara merekam beberapa peristiwa interaksi item dengan jenis acara yang berbeda dan nilai acara yang berbeda.

Saat Anda mengonfigurasi solusi, jika kumpulan data interaksi Item menyertakan bidang EVENT_TYPE dan EVENT_VALUE, Anda dapat menetapkan nilai tertentu sebagai ambang batas untuk mengecualikan catatan dari pelatihan. Untuk informasi selengkapnya, lihat Memilih data interaksi item yang digunakan untuk pelatihan.

Python

```
import boto3
import json

personalize_events = boto3.client(service_name='personalize-events')

personalize_events.put_events(
    trackingId = 'tracking_id',
    userId= 'user555',
    sessionId = 'session1',
    eventList = [{
        'eventId': 'event1',
        'sentAt': 1553631760,
        'eventType': 'like',
        'properties': json.dumps({
            'itemId': 'choc-panama',
            'eventValue': 4
            })
        }, {
        'eventId': 'event2',
        'sentAt': 1553631782,
        'eventType': 'rating',
        'properties': json.dumps({
            'itemId': 'movie_ten',
            'eventValue': 3
            })
        }]
)
```

AWS CLI

```
aws personalize-events put-events \
    --tracking-id tracking_id \
    --user-id user555 \
    --session-id session1 \
    --event-list '[{
        "eventId": "event1",
        "sentAt": 1553631760,
        "eventType": "like",
        "properties": "{\"itemId\": \"choc-panama\", \"eventValue\": \"true\"}"
      }, {
        "eventId": "event2",
        "sentAt": 1553631782,
        "eventType": "rating",
        "properties": "{\"itemId\": \"movie_ten\", \"eventValue\": \"4\",
 \"numRatings\": \"13\"}"
      }]'
```

SDK for Java 2.x

```
public static void putMultipleEvents(PersonalizeEventsClient
 personalizeEventsClient,
                             String trackingId,
                             String sessionId,
                             String userId,
                             String event1Type,
                             Float event1Value,
                             String event1ItemId,
                             int event1NumRatings,
                             String event2Type,
                             Float event2Value,
                             String event2ItemId,
                             int event2NumRatings) {

    ArrayList<Event> eventList = new ArrayList<Event>();

    try {
        Event event1 = Event.builder()
            .eventType(event1Type)
            .sentAt(Instant.ofEpochMilli(System.currentTimeMillis() + 10 * 60 *
 1000))
            .itemId(event1ItemId)
```

```
            .eventValue(event1Value)
            .properties("{\"numRatings\": "+ event1NumRatings +"}")
            .build();

        eventList.add(event1);

        Event event2 = Event.builder()
            .eventType(event2Type)
            .sentAt(Instant.ofEpochMilli(System.currentTimeMillis() + 10 * 60 *
 1000))
            .itemId(event2ItemId)
            .eventValue(event2Value)
            .properties("{\"numRatings\": "+ event2NumRatings +"}")
            .build();

        eventList.add(event2);

        PutEventsRequest putEventsRequest = PutEventsRequest.builder()
            .trackingId(trackingId)
            .userId(userId)
            .sessionId(sessionId)
            .eventList(eventList)
            .build();

        int responseCode = personalizeEventsClient.putEvents(putEventsRequest)
            .sdkHttpResponse()
            .statusCode();

        System.out.println("Response code: " + responseCode);

    } catch (PersonalizeEventsException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
}
```

> **Note**
>
> Kunci properti menggunakan nama kasus unta yang cocok dengan bidang dalam skema Interaksi. Misalnya, jika bidang 'NUM_RATINGS' didefinisikan dalam skema Interaksi, kunci properti harus. `numRatings`

Merekam peristiwa interaksi item dengan data tayangan

Jika Anda menggunakan Personalisasi Pengguna resep atau menambahkan bidang IMPRESSION ke skema untuk kumpulan data dalam grup kumpulan data Domain, Anda dapat merekam data tayangan dalam operasi Anda. PutEvents Tayangan adalah daftar item yang terlihat oleh pengguna saat berinteraksi dengan (misalnya, diklik atau ditonton) item tertentu. Amazon Personalize menggunakan data tayangan untuk memandu eksplorasi, di mana rekomendasi menyertakan item dengan data interaksi atau relevansi yang lebih sedikit. Untuk informasi tentang tayangan implisit dan eksplisit yang dapat dimodelkan Amazon Personalize, lihat. Data tayangan

> **Important**
>
> Jika Anda memberikan data tayangan implisit dan eksplisit yang bertentangan dalam `PutEvents` permintaan, Amazon Personalize menggunakan tayangan eksplisit secara default.

Untuk merekam rekomendasi Amazon Personalize yang Anda tampilkan kepada pengguna sebagai data tayangan, sertakan `recommendationId` dalam PutEvents permintaan Anda dan Amazon Personalize memperoleh tayangan implisit berdasarkan data rekomendasi Anda.

Untuk merekam data tayangan secara manual untuk suatu peristiwa, cantumkan tayangan dalam parameter `impression` input PutEvents perintah. Contoh kode berikut menunjukkan cara menyertakan `recommendationId` dan `impression` dalam PutEvents operasi dengan SDK for Python (Boto3) atau SDK for Java 2.x. Jika Anda menyertakan keduanya, Amazon Personalize menggunakan tayangan eksplisit secara default.

SDK for Python (Boto3)

```
import boto3

personalize_events = boto3.client(service_name='personalize-events')

personalize_events.put_events(
    trackingId = 'tracking_id',
    userId= 'userId',
    sessionId = 'sessionId',
    eventList = [{
        'eventId': 'event1',
        'eventType': 'rating',
```

```
        'sentAt': 1553631760,
        'itemId': 'item id',
        'recommendationId': 'recommendation id',
        'impression': ['itemId1', 'itemId2', 'itemId3']
        }]
)
```

SDK for Java 2.x

Gunakan `putEvents` metode berikut untuk merekam peristiwa dengan data tayangan dan RecommendationId. Untuk parameter tayangan, berikan daftar ItemIds sebagai file. ArrayList

```
public static void putEvents(PersonalizeEventsClient personalizeEventsClient,
                             String trackingId,
                             String sessionId,
                             String userId,
                             String eventType,
                             Float eventValue,
                             String itemId,
                             ArrayList<String> impressions,
                             String recommendationId) {

    try {
        Event event = Event.builder()
            .eventType(eventType)
            .sentAt(Instant.ofEpochMilli(System.currentTimeMillis() + 10 * 60 *
 1000))
            .itemId(itemId)
            .eventValue(eventValue)
            .impression(impressions)
            .recommendationId(recommendationId)
            .build();

        PutEventsRequest putEventsRequest = PutEventsRequest.builder()
            .trackingId(trackingId)
            .userId(userId)
            .sessionId(sessionId)
            .eventList(event)
            .build();

        int responseCode = personalizeEventsClient.putEvents(putEventsRequest)
            .sdkHttpResponse()
            .statusCode();
```

```
        System.out.println("Response code: " + responseCode);

    } catch (PersonalizeEventsException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
}
```

Metrik peristiwa dan laporan atribusi

Untuk memantau jenis dan jumlah peristiwa yang dikirim ke Amazon Personalize, gunakan metrik Amazon CloudWatch . Untuk informasi selengkapnya, lihat Memantau Amazon Personalisasi dengan Amazon CloudWatch.

Untuk menghasilkan CloudWatch laporan yang menunjukkan dampak rekomendasi, buat atribusi metrik dan catat interaksi pengguna dengan rekomendasi waktu nyata. Untuk informasi tentang membuat atribusi metrik, lihatMengukur dampak rekomendasi Amazon Personalize.

Untuk setiap acara, sertakan ID rekomendasi dari rekomendasi yang Anda tunjukkan kepada pengguna. Atau sertakan sumber acara, seperti pihak ketiga. Impor data ini untuk membandingkan berbagai kampanye, pemberi rekomendasi, dan pihak ketiga. Anda dapat mengimpor paling banyak 100 sumber atribusi acara.

- Jika Anda memberikan`recommendationId`, Amazon Personalize secara otomatis menentukan kampanye sumber atau pemberi rekomendasi dan mengidentifikasinya dalam laporan di kolom EVENT_ATTRIBUTION_SOURCE.
- Jika Anda memberikan kedua atribut tersebut, Amazon Personalize hanya menggunakan fitur. `eventAttributionSource`
- Jika Anda tidak menyediakan sumber, Amazon Personalize memberi label pada sumber `SOURCE_NAME_UNDEFINED` dalam laporan.

Kode berikut menunjukkan bagaimana menyediakan `eventAttributionSource` untuk suatu peristiwa dalam PutEvents operasi.

```
response = personalize_events.put_events(
    trackingId = 'eventTrackerId',
    userId= 'userId',
    sessionId = 'sessionId123',
    eventList = [{
```

```
        'eventId': 'event1',
        'eventType': 'watch',
        'sentAt': '1667260945',
        'itemId': '123',
        'metricAttribution': {
            'eventAttributionSource': 'thirdPartyServiceXYZ'
        }
    }]
)
statusCode = response['ResponseMetadata']['HTTPStatusCode']
print(statusCode)
```

Kode berikut menunjukkan bagaimana menyediakan `recommendationId` untuk suatu peristiwa dalam PutEvents operasi.

```
response = personalize_events.put_events(
    trackingId = 'eventTrackerId',
    userId= 'userId',
    sessionId = 'sessionId123',
    eventList = [{
        'eventId': 'event1',
        'eventType': 'watch',
        'sentAt': '1667260945',
        'itemId': '123',
        'recommendationId': 'RID-12345678-1234-1234-1234-abcdefghijkl'
    }]
)
statusCode = response['ResponseMetadata']['HTTPStatusCode']
print(statusCode)
```

Merekam peristiwa interaksi aksi waktu nyata

Peristiwa interaksi tindakan adalah interaksi antara pengguna dan tindakan. Misalnya, pengguna yang mendaftar dalam program keanggotaan atau mengajukan permohonan kartu kredit.

Jika Anda menggunakan resep kustom PERSONALIZED_ACTIONS, rekam peristiwa interaksi tindakan real-time saat pelanggan Anda berinteraksi dengan rekomendasi tindakan. Ini membangun data interaksi Anda dan membuat data Anda tetap segar. Ini juga memberi tahu Amazon Personalisasi tentang minat pengguna Anda saat ini, yang dapat meningkatkan relevansi rekomendasi. Hanya resep kustom PERSONALIZED_ACTIONS yang menggunakan data interaksi tindakan.

Anda merekam peristiwa interaksi tindakan dengan operasi PutActionInteractions API. Amazon Personalize menambahkan data ini ke kumpulan data interaksi Tindakan di grup kumpulan data Anda.

Peristiwa interaksi tindakan harus memiliki atribut tipe peristiwa, yang dapat berupa salah satu dari berikut ini:

- Diambil - Rekam peristiwa yang diambil saat pengguna mengambil tindakan yang disarankan.
- Tidak Diambil - Rekam peristiwa yang Tidak Diambil saat pengguna Anda membuat pilihan yang disengaja untuk tidak mengambil tindakan setelah melihatnya. Misalnya, jika mereka memilih Tidak ketika Anda menunjukkan tindakan kepada mereka. Peristiwa yang Tidak Diambil dapat menunjukkan bahwa pelanggan tidak tertarik dengan tindakan tersebut.
- Dilihat — Rekam peristiwa yang Dilihat saat Anda menunjukkan tindakan kepada pengguna sebelum mereka membuat pilihan untuk mengambil atau tidak mengambil tindakan. Amazon Personalize menggunakan Lihat peristiwa untuk mempelajari minat pengguna Anda. Misalnya, jika pengguna melihat tindakan tetapi tidak menerimanya, pengguna ini mungkin tidak tertarik dengan tindakan ini di masa mendatang.

Anda dapat merekam peristiwa real-time menggunakan AWS SDKs, atau AWS Command Line Interface (AWS CLI). Jika Anda merekam dua peristiwa dengan stempel waktu yang sama persis dan properti yang identik, Amazon Personalize hanya menyimpan salah satu acara.

Topik

- Persyaratan untuk merekam peristiwa interaksi tindakan
- Menemukan ID pelacak peristiwa interaksi tindakan Anda
- Merekam peristiwa interaksi aksi tunggal
- Merekam beberapa peristiwa interaksi aksi

Persyaratan untuk merekam peristiwa interaksi tindakan

Untuk merekam peristiwa interaksi aksi waktu nyata, Anda memerlukan yang berikut ini:

- Grup kumpulan data yang menyertakan`Action interactions dataset`, yang bisa kosong. Untuk informasi tentang membuat grup kumpulan data dan kumpulan data, lihat. Mengimpor data pelatihan ke Amazon Personalize dataset

- ID pelacak acara Anda. Anda menentukan ID ini dalam PutActionInteractions operasi. Saat Anda membuat kumpulan data interaksi Tindakan, Amazon Personalize secara otomatis membuat pelacak peristiwa interaksi tindakan untuk Anda. Untuk informasi selengkapnya, lihat Menemukan ID pelacak peristiwa interaksi tindakan Anda.
- Panggilan untuk PutActionInteractions operasi.

Menemukan ID pelacak peristiwa interaksi tindakan Anda

Saat Anda membuat kumpulan data interaksi Tindakan, Amazon Personalize secara otomatis membuat pelacak peristiwa interaksi tindakan untuk Anda. Anda menentukan ID pelacak dalam operasi PutActionInteractions API. Amazon Personalize menggunakannya untuk mengarahkan data baru ke kumpulan data interaksi Tindakan di grup kumpulan data Anda.

Anda dapat menemukan ID pelacak peristiwa di halaman detail kumpulan data interaksi Tindakan di konsol Amazon Personalize. Dan Anda dapat menemukan ID dengan memanggil operasi DescribeDataset API. Kode Python berikut mencetak ID pelacakan untuk kumpulan data interaksi Tindakan.

```
import boto3

personalize = boto3.client(service_name='personalize')

response = personalize.describe_dataset(
  datasetArn="Action interactions dataset ARN"
)

print(response['trackingId'])
```

Merekam peristiwa interaksi aksi tunggal

Setelah membuat kumpulan data Interaksi aksi, Anda siap merekam peristiwa interaksi tindakan dengan PutActionInteractions operasi tersebut. Kode berikut menunjukkan `PutActionInteractions` operasi yang melewati acara TAKEN. Anda dapat merekam acara ini ketika Anda menunjukkan rekomendasi pengguna dari Amazon Personalize dan mereka mengambil tindakan, seperti mengajukan permohonan kartu kredit Anda.

`actionInteractions`Ini adalah array ActionInteraction objek. Ini `trackingId` berasal dari pelacak peristiwa Amazon Personalize yang dibuat saat Anda membuat kumpulan data interaksi

Action. Untuk informasi selengkapnya, lihat Menemukan ID pelacak peristiwa interaksi tindakan Anda.

Aplikasi Anda menghasilkan unik `sessionId` ketika pengguna pertama kali mengunjungi situs web Anda atau menggunakan aplikasi Anda. Anda harus menggunakan hal yang sama `sessionId` di semua acara sepanjang sesi. Amazon Personalize menggunakan `sessionId` untuk mengaitkan peristiwa dengan pengguna sebelum mereka masuk (bersifat anonim). Untuk informasi selengkapnya, lihat Merekam acara untuk pengguna anonim.

`sentAt`Parameter`userId`,`actionId`, dan dipetakan ke bidang USER_ID, ACTION_ID, EVENT_TYPE, dan TIMESTAMP dari kumpulan data interaksi Tindakan.

Kumpulan data interaksi Tindakan yang sesuai

```
USER_ID, ACTION_ID, TIMESTAMP, EVENT_TYPE
user123, action-xyz, 1543631760, TAKEN
```

Contoh kode

AWS CLI

```
aws personalize-events put-action-interactions \
--tracking-id 12345678-xxxx-xxxx-xxxx-xxxxxxxxxxxx \
--action-interactions '[{
  "userId": "user123",
  "sessionId": "abcdefg",
  "timestamp": 1543631760,
  "eventType": "TAKEN",
  "actionId": "action-xyz"}]'
```

SDK for Python (Boto3)

```
import boto3

personalize_events = boto3.client(service_name='personalize-events')

response = personalize_events.put_action_interactions(
  trackingId='12345678-xxxx-xxxx-xxxx-xxxxxxxxxxxx',
  actionInteractions=[{
    'userId': 'user123',
    'sessionId': 'abcdefg',
    'timestamp': 1543631760,
```

```
    'eventType': 'Taken',
    'actionId': 'action-xyz'
  }]
)
```

Merekam beberapa peristiwa interaksi aksi

Kode berikut menunjukkan cara merekam beberapa peristiwa interaksi tindakan untuk pengguna yang sama dengan SessionId yang sama.

Kumpulan data interaksi Tindakan yang sesuai

```
USER_ID, ACTION_ID, EVENT_TYPE, TIMESTAMP
user123, action123, Taken, 1543531139
user123, action345, Not Taken, 1543531139
```

AWS CLI

```
aws personalize-events put-action-interactions \
--tracking-id 6ddfe6b7-cd83-4dd4-b09d-4c35ecbacfe1 \
--action-interactions '[{
  "userId": "user123",
  "sessionId": "abcdefg",
  "timestamp": 1543531139,
  "eventType": "Taken",
  "actionId": "action123"
},
{
  "userId": "user123",
  "sessionId": "abcdefg",
  "timestamp": 1543531139,
  "eventType": "Not Taken",
  "actionId": "action345"}]'
```

SDK for Python (Boto3)

```
import boto3

personalize_events = boto3.client(service_name='personalize-events')

response = personalize_events.put_action_interactions(
```

```
  trackingId='12345678-xxxx-xxxx-xxxx-xxxxxxxxxxxx',
  actionInteractions=[{
    'userId': 'user123',
    'sessionId': 'abcdefg',
    'timestamp': 1697848587,
    'eventType': 'Taken',
    'actionId': 'action123'
  },
  {
    'userId': 'user123',
    'sessionId': 'abcdefg',
    'timestamp': 1697848622,
    'eventType': 'Not Taken',
    'actionId': 'action345'
  }]
)
```

Merekam acara untuk pengguna anonim

> **Important**
>
> Jika Anda tidak merekam setidaknya satu peristiwa dengan `sessionId` dan `userId` untuk pengguna, Amazon Personalize tidak akan menggunakan aktivitas yang dilacak hanya pada saat `sessionId` latihan. Dan setelah pelatihan selesai, rekomendasi tidak akan lagi didasarkan pada aktivitas yang dilacak ke. `sessionId` Ini akan membuat riwayat acara berkelanjutan untuk userIds sebelum dan sesudah mereka masuk.

Anda dapat merekam interaksi item atau peristiwa interaksi tindakan untuk pengguna sebelum mereka membuat akun. Rekam peristiwa untuk pengguna anonim untuk membangun riwayat acara berkelanjutan dengan peristiwa dari sebelum dan sesudah mereka masuk. Ini memberikan Amazon Personalisasi lebih banyak data interaksi tentang pengguna, yang dapat membantu menghasilkan rekomendasi yang lebih relevan.

Untuk merekam peristiwa untuk pengguna anonim (pengguna yang belum masuk), untuk setiap peristiwa, tentukan hanya `asessionId`. Aplikasi Anda menghasilkan unik `sessionId` ketika pengguna pertama kali mengunjungi situs web Anda atau menggunakan aplikasi Anda. Anda harus menggunakan hal yang sama `sessionId` di semua acara sepanjang sesi. Amazon Personalize menggunakan `sessionId` untuk mengaitkan peristiwa dengan pengguna sebelum mereka masuk.

Amazon Personalize tidak menggunakan peristiwa dari pengguna anonim saat berlatih hingga Anda mengaitkannya dengan file. `userId` Untuk informasi selengkapnya, lihat Membangun riwayat acara berkelanjutan untuk pengguna anonim.

Untuk memberikan personalisasi real-time bagi pengguna anonim, tentukan sessionID sebagai userID dalam permintaan atau Anda. GetRecommendations GetActionRecommendations

- Untuk contoh kode yang menunjukkan cara merekam peristiwa interaksi item dengan PutEvents operasi dan SessionId dan userID, lihat. Merekam peristiwa interaksi item tunggal
- Untuk contoh kode yang menunjukkan cara merekam peristiwa interaksi tindakan dengan PutActionInteractions operasi dan SessionId serta userId, lihat. Merekam peristiwa interaksi aksi tunggal

Membangun riwayat acara berkelanjutan untuk pengguna anonim

Untuk membuat riwayat acara untuk pengguna anonim dan meminta Amazon Personalisasi menggunakan acara mereka saat pelatihan, rekam setidaknya satu acara dengan a `sessionId` dan a. `userId` Kemudian Anda dapat merekam sejumlah acara untuk`userId`. Setelah Anda mulai memberikan`userId`, `sessionId` kaleng berubah. Selama pelatihan ulang penuh berikutnya, Amazon Personalize mengaitkannya `userId` dengan riwayat pengguna anonim yang dilacak ke aslinya. `sessionId`

Setelah pelatihan ulang selesai, rekomendasi akan didasarkan pada aktivitas yang dilacak baik `sessionId` dari peristiwa anonim maupun acara apa pun yang dilacak ke acara mereka. `userId`

> **Note**
>
> Jika pengguna Anda tidak membuat akun dan Anda ingin Amazon Personalize menggunakan data saat pelatihan, Anda dapat menggunakan `sessionId` as the `userId` in events. Namun, jika pengguna akhirnya membuat akun, Anda tidak akan dapat mengaitkan peristiwa dari penjelajahan anonim mereka dengan yang baru`userId`.

Layanan pelacakan acara pihak ketiga

Platform Data Pelanggan berikut (CDPs) dapat membantu Anda mengumpulkan data peristiwa dari aplikasi Anda dan mengirimkannya ke Amazon Personalize.

- Amplitudo — Anda dapat menggunakan Amplitudo untuk melacak tindakan pengguna untuk membantu Anda memahami perilaku pengguna Anda. Untuk informasi tentang penggunaan Amplitude dan Amazon Personalize, lihat postingan blog Partner Network (APN) AWS berikut: Mengukur Efektivitas Personalisasi dengan Amplitudo dan Amazon Personalize.
- Segmen — Anda dapat menggunakan Segmen untuk mengirim data Anda ke Amazon Personalize. Untuk informasi selengkapnya tentang mengintegrasikan Segmen dengan Amazon Personalize, lihat Amazon Personalize Destination.

Contoh implementasi

Untuk contoh notebook Jupyter yang menunjukkan cara menggunakan Amazon Personalize untuk bereaksi terhadap perilaku real-time pengguna menggunakan pelacak peristiwa dan operasi, lihat PutEvents 2.VIEW_CAMPAIGN_AND_INTERACTIONS.IPynb di folder getting_started repositori. amazon-personalize-samples GitHub

Untuk contoh yang menunjukkan cara melakukan streaming peristiwa dari pengguna yang berinteraksi dengan rekomendasi, lihat streaming_events di repositori Amazon Personalize samples. GitHub

Untuk contoh lengkap yang berisi kode sumber dan file pendukung untuk digunakan secara real-time APIs yang berada di antara sumber daya Amazon Personalisasi dan aplikasi klien, lihat Personalisasi Waktu Nyata di APIs AWS repositori sampel. GitHub Proyek ini mencakup bagaimana menerapkan hal-hal berikut:

- Konteks pengguna dan koleksi acara pengguna
- Caching respons
- Rekomendasi dekorasi berdasarkan metadata item
- Pengujian A/B
- Otentikasi API

Mempertahankan relevansi rekomendasi

Rekomendasi yang relevan dapat meningkatkan keterlibatan pengguna, rasio klik-tayang, dan rasio konversi untuk aplikasi Anda seiring pertumbuhan katalog Anda. Untuk mempertahankan dan meningkatkan relevansi rekomendasi Amazon Personalisasi bagi pengguna Anda, perbarui data dan sumber daya khusus Anda. Hal ini memungkinkan Amazon Personalize untuk belajar dari perilaku terbaru pengguna Anda dan menyertakan item terbaru Anda dalam rekomendasi.

Topik

- Menjaga set data saat ini
- Mempertahankan rekomendasi domain
- Mempertahankan solusi khusus

Menjaga set data saat ini

Seiring pertumbuhan katalog Anda, perbarui data historis Anda dengan operasi impor data massal atau individual. Untuk informasi selengkapnya tentang mengimpor data historis, lihatMengimpor data pelatihan ke Amazon Personalize dataset. Untuk informasi tentang cara data yang Anda impor setelah melatih model memengaruhi rekomendasi, lihatMemperbarui data dalam kumpulan data setelah pelatihan.

Untuk kasus penggunaan dan resep yang memberikan rekomendasi real-time yang dipersonalisasi, perbarui kumpulan data interaksi Item Anda dengan perilaku pengguna Anda. Lakukan ini dengan merekam interaksi item dengan pelacak peristiwa dan operasi PutEvents API. Amazon Personalisasi pembaruan rekomendasi berdasarkan aktivitas terbaru pengguna Anda saat berinteraksi dengan katalog Anda. Untuk informasi tentang personalisasi waktu nyata, lihatPersonalisasi waktu nyata. Untuk informasi selengkapnya tentang merekam peristiwa waktu nyata, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi.

Mempertahankan rekomendasi domain

Amazon Personalize secara otomatis melatih ulang model yang mendukung rekomendasi Anda setiap 7 hari. Ini adalah pelatihan ulang penuh yang membuat model yang sama sekali baru berdasarkan keseluruhan data dalam kumpulan data Anda. Jika Anda memodifikasi kolom yang digunakan dalam pelatihan, Amazon Personalize secara otomatis memulai pelatihan ulang penuh model yang mendukung pemberi rekomendasi Anda.

- Untuk Pilihan teratas untuk Anda dan Direkomendasikan untuk kasus penggunaan Anda, Amazon Personalisasi memperbarui pemberi rekomendasi Anda untuk mempertimbangkan item baru untuk rekomendasi. Pembaruan otomatis bukanlah pelatihan ulang penuh di mana model belajar dari perilaku pengguna Anda. Sebagai gantinya, pembaruan otomatis memungkinkan Amazon Personalisasi untuk menampilkan item baru Anda dalam rekomendasi sebelum pelatihan ulang penuh pemberi rekomendasi berikutnya. Untuk informasi tentang pembaruan otomatis, lihatPembaruan otomatis.
- Jika Anda menggunakan kasus penggunaan Trending now, Amazon Personalize secara otomatis mengevaluasi data interaksi Anda setiap dua jam dan mengidentifikasi item yang sedang tren. Anda tidak perlu menunggu rekomendasi Anda untuk berlatih kembali.

Sementara pelatihan ulang pemberi rekomendasi sedang berlangsung, Anda masih bisa mendapatkan rekomendasi dari pemberi rekomendasi. Sampai pelatihan ulang selesai, pemberi rekomendasi menggunakan konfigurasi dan model sebelumnya. Untuk melacak pembaruan, Anda dapat melihat stempel waktu untuk pembaruan pemberi rekomendasi terbaru di halaman Detail Recommender di konsol Amazon Personalize. Atau Anda dapat melihat `latestRecommenderUpdate` detail dari DescribeRecommender operasi.

Mempertahankan solusi khusus

Secara default, semua solusi baru menggunakan pelatihan otomatis untuk membuat versi solusi baru setiap 7 hari. Pelatihan berlanjut sampai Anda menghapus solusinya.

Saat Anda membuat solusi, kami sarankan Anda menggunakan pelatihan otomatis untuk mengelola pembuatan versi solusi. Ini membuat pemeliharaan solusi Anda lebih mudah. Ini menghapus pelatihan manual yang diperlukan untuk solusi untuk belajar dari data Anda yang lebih baru. Tanpa pelatihan otomatis, Anda harus membuat versi solusi baru secara manual agar solusi dapat dipelajari dari data terbaru Anda. Untuk informasi selengkapnya tentang mengonfigurasi pelatihan otomatis, lihatMengkonfigurasi pelatihan otomatis.

Frekuensi pelatihan Anda tergantung pada kebutuhan bisnis Anda, resep yang Anda gunakan, dan seberapa sering Anda mengimpor data. Untuk semua resep, kami merekomendasikan pelatihan setidaknya setiap minggu. Dengan pelatihan otomatis, ini adalah frekuensi pelatihan default. Jika Anda sering menambahkan item atau tindakan baru, Anda mungkin ingin memiliki frekuensi pelatihan yang lebih tinggi, tergantung pada resep Anda.

- Jika Anda menggunakan User-Personalization-v 2, Personalisasi Pengguna, atau Next-Best-Action, solusi akan diperbarui secara otomatis untuk mempertimbangkan item atau tindakan baru untuk rekomendasi. Pembaruan otomatis tidak sama dengan pelatihan otomatis. Pembaruan otomatis tidak membuat versi solusi yang sama sekali baru, dan model tidak belajar dari data terbaru Anda. Untuk mempertahankan solusi Anda, frekuensi pelatihan Anda harus tetap setidaknya setiap minggu. Untuk informasi selengkapnya tentang pembaruan otomatis, termasuk pedoman dan persyaratan tambahan, lihatPembaruan otomatis.
- Jika Anda menggunakan Trending-Now, Amazon Personalize secara otomatis mengidentifikasi item tren teratas dalam data interaksi Anda selama interval waktu yang dapat dikonfigurasi. Trending-Now dapat merekomendasikan item yang ditambahkan sejak pelatihan terakhir melalui data interaksi massal atau streaming. Frekuensi pelatihan Anda harus tetap setidaknya setiap minggu. Untuk informasi selengkapnya, lihat Resep Trend-Now.
- Jika Anda tidak menggunakan resep dengan pembaruan otomatis atau resep Trending-Now, Amazon Personalize mempertimbangkan item baru untuk rekomendasi hanya setelah pelatihan berikutnya. Misalnya, jika Anda menggunakan resep Barang Serupa, dan Anda menambahkan item baru setiap hari, Anda harus menggunakan frekuensi pelatihan harian agar item ini muncul dalam rekomendasi pada hari yang sama.

Memperbarui data dalam kumpulan data setelah pelatihan

Seiring pertumbuhan katalog Anda, impor data pelatihan tambahan ke dalam kumpulan data Anda. Ini membantu menjaga dan meningkatkan relevansi rekomendasi Amazon Personalize. Anda dapat mengimpor lebih banyak data dengan operasi impor data massal atau individual.

- Dengan impor individual, Amazon Personalize menambahkan catatan baru ke kumpulan data. Untuk memperbarui item, pengguna, atau tindakan individual, Anda dapat mengimpor rekaman dengan ID yang sama tetapi dengan atribut yang dimodifikasi. Anda dapat mengimpor hingga 10 catatan per operasi impor individual.

  Untuk informasi selengkapnya tentang mengimpor catatan secara individual, lihatMengimpor catatan individual ke dalam kumpulan data Amazon Personalize. Untuk informasi tentang merekam peristiwa waktu nyata, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi.
- Dengan impor massal, Anda menambah atau mengganti data massal dengan membuat pekerjaan impor lain. Secara default, pekerjaan impor dataset menggantikan data yang ada dalam kumpulan data yang Anda impor secara massal. Sebagai gantinya, Anda dapat menambahkan catatan baru ke data yang ada dengan mengubah mode impor pekerjaan.

  Untuk menambahkan data ke kumpulan data interaksi Item atau Kumpulan data interaksi tindakan dengan pekerjaan impor kumpulan data, Anda harus memiliki minimal 1000 interaksi item baru atau catatan interaksi tindakan. Dalam waktu 20 menit setelah menyelesaikan impor massal, Amazon Personalize memperbarui filter apa pun yang Anda buat di grup kumpulan data dengan data massal baru Anda. Pembaruan ini memungkinkan Amazon Personalize untuk menggunakan data terbaru saat memfilter rekomendasi untuk pengguna Anda.

Setelah membuat kumpulan data Item atau Pengguna, Anda dapat mengganti skema dengan yang baru atau yang sudah ada. Anda dapat mengganti skema kumpulan data jika struktur data Anda berubah setelah Anda membuat kumpulan data. Misalnya, Anda mungkin memiliki kolom metadata item baru yang ingin dipertimbangkan Amazon Personalize selama pelatihan. Atau Anda mungkin ingin menambahkan kolom data untuk digunakan hanya saat memfilter rekomendasi. Untuk informasi selengkapnya, lihat Mengganti skema dataset untuk menambahkan kolom baru.

Setelah Anda membuat versi rekomendasi atau solusi kustom, bagaimana data baru memengaruhi rekomendasi bergantung pada jenisnya, metode impor, dan kasus penggunaan domain atau resep khusus yang Anda gunakan. Bagian berikut menjelaskan bagaimana data baru memengaruhi rekomendasi real-time dan batch sebelum pelatihan berikutnya.

Topik

- Bagaimana data baru memengaruhi rekomendasi waktu nyata
- Bagaimana data baru memengaruhi rekomendasi batch (sumber daya khusus)

Bagaimana data baru memengaruhi rekomendasi waktu nyata

Setelah Anda membuat versi rekomendasi atau solusi kustom, bagaimana data baru memengaruhi rekomendasi waktu nyata bergantung pada jenis data, metode impor, dan kasus penggunaan domain atau resep khusus yang Anda gunakan. Bagian berikut menjelaskan bagaimana data baru memengaruhi rekomendasi waktu nyata sebelum pelatihan berikutnya.

Pelatihan dapat berupa pelatihan otomatis mingguan pemberi rekomendasi, atau pembuatan versi solusi otomatis atau manual. Untuk pelatihan manual dengan User-Personalization, hilangkan `trainingMode` untuk menggunakan mode pelatihan default. `FULL`

Topik

- Interaksi baru
- Item baru
- Pengguna baru
- Tindakan baru

Interaksi baru

Interaksi baru adalah interaksi item atau tindakan yang Anda impor setelah pelatihan terbaru. Untuk data real-time dan massal, jika interaksi melibatkan item atau tindakan baru, Amazon Personalize dapat mempertimbangkannya untuk rekomendasi tanpa pelatihan jika resep atau kasus penggunaan Anda menampilkan eksplorasi. Untuk informasi selengkapnya, lihat Item baru atau Tindakan baru.

Acara waktu nyata

Untuk kasus penggunaan dan resep yang menampilkan personalisasi waktu nyata, Amazon Personalize segera menggunakan interaksi real-time antara pengguna dan item atau tindakan yang ada di pelatihan terbaru. Saat membuat rekomendasi untuk pengguna di ventilasi, Amazon Personalize menggunakan interaksi real-time ini. Untuk informasi selengkapnya tentang personalisasi waktu nyata, lihatPersonalisasi waktu nyata.

Untuk kasus penggunaan domain dan resep khusus yang tidak menampilkan personalisasi waktu nyata, seperti merekomendasikan item serupa, model Anda belajar dari data interaksi waktu nyata hanya setelah pelatihan.

Interaksi massal

Untuk interaksi massal, untuk pekerjaan impor kumpulan data tambahan dan penuh, model Anda belajar dari interaksi item massal atau data interaksi tindakan hanya setelah pelatihan berikutnya. Data massal tidak digunakan untuk memperbarui rekomendasi untuk personalisasi waktu nyata.

Untuk informasi selengkapnya tentang mengimpor lebih banyak data massal, lihatMengimpor data massal ke Amazon Personalisasi dengan pekerjaan impor kumpulan data.

Item baru

Item baru adalah item yang Anda impor setelah pelatihan terbaru. Mereka dapat berasal dari data interaksi atau metadata item dalam kumpulan data Item.

Item baru dipertimbangkan untuk rekomendasi sebagai berikut:

- Untuk Pilihan teratas untuk Anda dan Direkomendasikan untuk kasus domain atau User-Personalization-v 2 kasus, Personalisasi Pengguna, atau resep, Next-Best-Action Amazon Personalize secara otomatis memperbarui model setiap dua jam. Setelah setiap pembaruan, Amazon Personalize mempertimbangkan item baru untuk rekomendasi sebagai bagian dari eksplorasi. Saat mempertimbangkan item baru, Amazon Personalize mempertimbangkan metadata apa pun untuk item tersebut. Namun data ini akan memiliki efek yang lebih besar pada rekomendasi hanya setelah Anda merekam interaksi untuk item dan melatih model baru. Untuk informasi tentang pembaruan, lihatPembaruan otomatis.
- Jika Anda menggunakan kasus penggunaan Trending now, Amazon Personalize secara otomatis mengevaluasi data interaksi Anda setiap dua jam dan mengidentifikasi item yang sedang tren. Anda tidak perlu menunggu rekomendasi Anda untuk berlatih. Jika Anda menggunakan resep Trending-Now, Amazon Personalize secara otomatis mempertimbangkan semua item baru selama interval yang dapat dikonfigurasi tanpa pelatihan. Untuk informasi tentang mengkonfigurasi interval, lihatResep Trend-Now.
- Jika Anda tidak menggunakan resep Trending-Now atau kasus penggunaan atau resep Anda tidak mendukung pembaruan otomatis, Amazon Personalize akan mempertimbangkan item baru hanya setelah pelatihan berikutnya.

Pengguna baru

Pengguna baru adalah pengguna yang Anda impor setelah pelatihan terbaru. Mereka dapat berasal dari data interaksi atau metadata pengguna dalam kumpulan data Pengguna. Untuk pengguna baru dan anonim (pengguna tanpa UserID), Anda dapat merekam peristiwa untuk pengguna dengan `sessionId` dan Amazon Personalize akan mengaitkan peristiwa dengan pengguna sebelum mereka masuk. Untuk informasi selengkapnya, lihat Merekam acara untuk pengguna anonim.

Amazon Personalize menghasilkan rekomendasi untuk pengguna baru sebagai berikut:

- Jika Anda menggunakan kasus penggunaan domain Trending sekarang atau resep kustom Trending-Now, pengguna baru segera menerima rekomendasi untuk item tren teratas. Jika Anda menggunakan resep Popularity-Count, pengguna baru segera menerima rekomendasi untuk item dengan interaksi terbanyak.
- Untuk resep atau kasus penggunaan yang memberikan rekomendasi yang dipersonalisasi untuk pengguna, rekomendasi untuk pengguna baru didasarkan pada riwayat interaksi awal pengguna Anda yang ada. Item atau tindakan pertama yang berinteraksi dengan pengguna yang ada ini lebih mungkin direkomendasikan kepada pengguna baru. Untuk resep Personalisasi Pengguna atau Peringkat Personalisasi, jika Anda `recency_mask` menyetelnya`true`, rekomendasi juga menyertakan item berdasarkan tren popularitas terbaru dalam data interaksi Anda.

Berikut ini dapat meningkatkan relevansi rekomendasi untuk pengguna baru:

- Data interaksi — Cara utama untuk meningkatkan relevansi rekomendasi bagi pengguna baru adalah dengan mengimpor data dari interaksi mereka dengan item Anda. Untuk informasi tentang bagaimana data interaksi baru memengaruhi rekomendasi, lihatInteraksi baru.
- Metadata pengguna — Mengimpor metadata pengguna, seperti GENDER atau MEMBERSHIP_STATUS, dapat meningkatkan rekomendasi. Agar metadata memengaruhi rekomendasi, Anda harus menunggu pelatihan ulang otomatis mingguan pemberi rekomendasi domain Anda selesai. Atau Anda harus secara manual membuat versi solusi baru.
- Metadata kontekstual — Jika kasus penggunaan atau resep Anda mendukung metadata kontekstual dan kumpulan data interaksi Item Anda memiliki bidang metadata untuk data kontekstual, Anda dapat memberikan konteks pengguna dalam permintaan rekomendasi Anda. Ini tidak memerlukan pelatihan ulang. Untuk informasi selengkapnya, lihat Meningkatkan relevansi rekomendasi dengan metadata kontekstual.

Tindakan baru

Tindakan baru adalah tindakan yang Anda impor sejak pelatihan terbaru. Mereka dapat berasal dari data interaksi tindakan atau tindakan dalam kumpulan data Tindakan.

Dengan Next-Best-Action resepnya, Amazon Personalize secara otomatis memperbarui versi solusi setiap dua jam. Setelah setiap pembaruan, Amazon Personalize mempertimbangkan tindakan baru untuk rekomendasi sebagai bagian dari eksplorasi. Saat mempertimbangkan tindakan baru, Amazon Personalize mempertimbangkan metadata apa pun untuk tindakan tersebut. Namun, data ini akan memiliki efek yang lebih besar pada rekomendasi hanya setelah Anda merekam interaksi tindakan untuk tindakan dan melatih ulang sepenuhnya. Untuk informasi tentang pembaruan, lihat Pembaruan otomatis

Bagaimana data baru memengaruhi rekomendasi batch (sumber daya khusus)

Setelah Anda membuat versi solusi kustom, bagaimana data baru memengaruhi rekomendasi batch bergantung pada jenis data, metode impor, dan resep kustom yang Anda gunakan.

Untuk segmen pengguna, Amazon Personalize menghasilkan segmen hanya menggunakan data yang ada pada pelatihan versi solusi lengkap terakhir. Dan Amazon Personalize hanya menggunakan data massal yang Anda impor dengan mode impor FULL (menggantikan data yang ada). Untuk informasi selengkapnya tentang segmen pengguna, lihatMendapatkan segmen pengguna batch dengan sumber daya khusus.

Saat membuat rekomendasi item batch, Amazon Personalize mempertimbangkan semua data massal yang ada pada saat pembuatan versi solusi terbaru. Data ini dapat diimpor dengan mode impor FULL atau INCREMENTAL. Agar catatan massal yang lebih baru dapat memengaruhi rekomendasi batch, Anda harus membuat versi solusi baru dan kemudian membuat pekerjaan inferensi batch.

Bagian berikut menjelaskan bagaimana impor individu memengaruhi rekomendasi item batch.

Topik

- Interaksi baru
- Pengguna baru
- Item baru

Interaksi baru

Jika Anda menggunakan resep USER_PERSONALIZATION atau PERSONALIZED_RANKING, Amazon Personalize mempertimbangkan data interaksi item baru dengan item dan pengguna yang ada dalam waktu sekitar 15 menit dari impor data. Item dan pengguna ini pasti telah hadir pada pelatihan terbaru. Untuk memastikan acara dipertimbangkan, kami sarankan Anda menunggu minimal 15 menit sebelum memulai pekerjaan inferensi batch. Untuk semua resep lain, dan untuk acara dengan item atau pengguna baru, Anda harus membuat versi solusi baru untuk acara streaming untuk memengaruhi rekomendasi batch.

Pengguna baru

Untuk pengguna tanpa data interaksi, rekomendasi awalnya hanya untuk item populer. Jika Anda menggunakan resep USER_PERSONALIZATION atau PERSONALIZED_RANKING dan Anda merekam peristiwa untuk pengguna, rekomendasi mereka mungkin menjadi lebih relevan dalam waktu sekitar 15 menit setelah impor tanpa pelatihan ulang. Untuk memastikan acara dipertimbangkan, kami sarankan Anda menunggu minimal 15 menit sebelum memulai pekerjaan inferensi batch. Untuk semua resep lainnya, Anda harus membuat versi solusi baru untuk acara streaming guna memengaruhi rekomendasi batch bagi pengguna tanpa data interaksi.

Item baru

Dengan User-Personalization-v 2 dan Personalisasi Pengguna, saat Anda membuat pekerjaan inferensi batch dan menentukan versi solusi terlatih lengkap terbaru untuk solusi Anda, Amazon Personalize secara otomatis memperbarui versi solusi untuk menyertakan item baru dalam rekomendasi dengan eksplorasi. Jika Anda tidak menentukan versi solusi terbaru, tidak ada pembaruan yang terjadi. Untuk resep lainnya, Anda harus membuat versi solusi baru untuk item baru yang akan ditampilkan dalam rekomendasi batch. Untuk informasi lebih lanjut tentang eksplorasi, lihatEksplorasi.

Mengganti skema dataset untuk menambahkan kolom baru

Setelah membuat kumpulan data Item atau Pengguna, Anda dapat mengganti skema dengan yang baru atau yang sudah ada. Anda dapat mengganti skema kumpulan data jika struktur data Anda berubah setelah Anda membuat kumpulan data. Misalnya, Anda mungkin memiliki kolom metadata item baru yang ingin dipertimbangkan Amazon Personalize selama pelatihan. Atau Anda mungkin ingin menambahkan kolom data untuk digunakan hanya saat memfilter rekomendasi.

Saat mengganti skema kumpulan data, Anda harus menyimpan semua bidang dalam skema sebelumnya dan Anda tidak dapat mengubah tipe atau atribut datanya. Setelah Anda mengganti skema kumpulan data, Amazon Personalize secara otomatis mengecualikan kolom baru apa pun dari pelatihan untuk rekomendasi atau solusi khusus yang ada. Untuk panduan dan persyaratan lebih lanjut, lihatPedoman dan persyaratan.

Anda dapat mengganti skema kumpulan data dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), dan. AWS SDKs

Topik

- Pedoman dan persyaratan
- Mengganti skema kumpulan data (konsol)
- Mengganti skema () kumpulan data AWS CLI
- Mengganti skema () kumpulan data AWS SDKs

Pedoman dan persyaratan

Sebelum Anda mengganti skema untuk kumpulan data, pastikan Anda mengetahui pedoman dan persyaratan berikut:

- Anda tidak dapat mengganti skema kumpulan data interaksi Item, kumpulan data interaksi tindakan, atau kumpulan data Tindakan.
- Anda dapat menambahkan bidang baru ke skema pengganti Anda, tetapi Anda harus menyimpan semua bidang dalam skema sebelumnya. Dan Anda tidak dapat mengubah tipe atau atribut data mereka. Misalnya, jika skema sebelumnya menyertakan `MEMBERSHIP_STATUS` bidang untuk data string kategoris, skema baru yang Anda gunakan harus menyertakan `MEMBERSHIP_STATUS` bidang dengan atribut dan tipe data ini.

- Jika skema saat ini memiliki bidang yang ingin Anda ganti namanya, atau jika Anda ingin mengubah tipe atau atribut datanya, Anda dapat menambahkan bidang baru dengan nama baru dan jenis atau atribut yang dimodifikasi. Kemudian sertakan bidang baru dalam pelatihan dan kecualikan bidang lama. Setiap bidang baru harus mendukung `null` data. Jika bidang lama tidak mendukung data null, saat Anda mengimpor data, Anda dapat menggunakan data placeholder untuk memastikan impor Anda cocok dengan skema. Untuk informasi tentang mengonfigurasi kolom yang digunakan oleh pemberi rekomendasi, lihat. Memperbarui pemberi rekomendasi Untuk informasi tentang mengonfigurasi kolom yang digunakan oleh solusi, lihatMengkonfigurasi kolom yang digunakan saat pelatihan.
- Setiap bidang baru harus mendukung `null` data. Untuk informasi tentang menambahkan tipe null ke bidang, lihatJenis data skema.
- Setelah Anda mengganti skema kumpulan data, Amazon Personalize secara otomatis mengecualikan kolom baru apa pun dari pelatihan untuk rekomendasi atau solusi khusus yang ada. Menggunakan dataset yang dimodifikasi melibatkan tindakan berikut:
  - Untuk menggunakan kolom baru apa pun dalam pelatihan, impor data yang selaras dengan skema baru. Kemudian perbarui rekomendasi apa pun untuk menggunakan kolom baru, atau buat solusi khusus baru dan konfigurasikan kolom yang digunakannya saat pelatihan.

    Untuk informasi tentang memperbarui kolom yang digunakan oleh pemberi rekomendasi, lihatMemperbarui pemberi rekomendasi. Untuk informasi tentang mengonfigurasi kolom yang digunakan oleh solusi, lihatMengkonfigurasi kolom yang digunakan saat pelatihan.
  - Untuk menggunakan kolom apa pun hanya saat memfilter, impor data yang sejajar dengan skema baru, buat filter yang menggunakan data baru, dan terapkan filter Anda ke permintaan rekomendasi Anda. Anda tidak perlu memperbarui pemberi rekomendasi apa pun, atau membuat atau memperbarui sumber daya khusus apa pun.

Mengganti skema kumpulan data (konsol)

Untuk mengganti skema kumpulan data dengan konsol Amazon Personalize, Anda memilih kumpulan data yang akan dimodifikasi dan memilih untuk mengganti dengan skema baru atau menggunakan skema yang sudah ada.

Untuk mengganti skema dataset

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.

2. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
3. Di panel navigasi, pilih Datasets, dan pilih tombol radio untuk dataset yang ingin Anda modifikasi.
4. Pilih Tindakan, dan pilih Ganti skema.
5. Dalam detail Skema, pilih untuk mengganti dengan skema baru atau skema yang dibuat sebelumnya.
6. Tentukan skema baru yang akan digunakan. Jika Anda telah memilih untuk:

   - Ganti dengan skema baru, lalu beri nama skema, dan dalam definisi Skema, buat perubahan Anda pada skema JSON.
   - Gunakan skema yang dibuat sebelumnya, lalu untuk skema yang dibuat sebelumnya, pilih skema yang ingin Anda gunakan. Hanya skema yang memenuhi syarat yang terdaftar. Untuk informasi tentang persyaratan skema, lihatPedoman dan persyaratan.
7. Pilih Ganti. Saat kumpulan data aktif, Anda dapat mulai mengimpor data yang selaras dengan skema baru. Untuk informasi selengkapnya, lihat Mengimpor data pelatihan ke Amazon Personalize dataset.

Mengganti skema () kumpulan data AWS CLI

Untuk mengganti skema kumpulan data dengan AWS CLI, Anda menggunakan `update-dataset` perintah, tentukan Nama Sumber Daya Amazon (ARN) dari kumpulan data yang akan diperbarui dan ARN skema baru yang akan digunakan. Anda tidak dapat memperbarui skema kumpulan data interaksi Item, kumpulan data interaksi tindakan, atau kumpulan data Tindakan.

Kode berikut menunjukkan cara memperbarui skema kumpulan data dengan. AWS CLI Untuk mengganti skema dataset dengan yang baru, pertama-tama gunakan perintah. `create-schema` Kemudian gunakan kode berikut untuk mengganti skema saat ini dengan yang baru. Untuk informasi tentang membuat skema dengan AWS CLI, lihatMembuat dataset dan skema ()AWS CLI. Untuk informasi tentang kumpulan data dan persyaratan skema, lihat. Membuat file JSON skema untuk skema Amazon Personalize

```
aws personalize update-dataset \
--dataset-arn Dataset ARN \
--schema-arn New schema ARN
```

Saat kumpulan data aktif, Anda dapat mulai mengimpor data yang selaras dengan skema baru. Untuk informasi selengkapnya, lihat Mengimpor data pelatihan ke Amazon Personalize dataset.

Untuk informasi tentang pembaruan terbaru ke kumpulan data, Anda dapat menggunakan DescribeDataset operasi ini.

Mengganti skema () kumpulan data AWS SDKs

Untuk mengganti skema kumpulan data dengan AWS SDKs, Anda menggunakan operasi `UpdateDataset` API. Tentukan Nama Sumber Daya Amazon (ARN) dari kumpulan data yang akan diperbarui dan skema baru yang akan digunakan. Anda tidak dapat memperbarui skema kumpulan data interaksi Item, kumpulan data interaksi tindakan, atau kumpulan data Tindakan.

Kode berikut menunjukkan cara mengganti skema dataset dengan SDK for Python (Boto3). Untuk mengganti skema dataset dengan yang baru, pertama gunakan operasi. CreateSchema Kemudian gunakan kode berikut untuk mengganti skema saat ini dengan yang baru. Untuk informasi tentang membuat skema dengan AWS SDKs, lihatMembuat dataset dan skema ()AWS SDKs. Untuk informasi tentang set data dan persyaratan skema, lihat. Membuat file JSON skema untuk skema Amazon Personalize

```
import boto3

personalize = boto3.client('personalize')

update_dataset_response = personalize.update_dataset(
    datasetArn = 'dataset_arn',
    schemaArn = 'new_schema_arn'
)

print(update_dataset_response)
```

Saat kumpulan data aktif, Anda dapat mulai mengimpor data yang selaras dengan skema baru. Untuk informasi selengkapnya, lihat Mengimpor data pelatihan ke Amazon Personalize dataset. Untuk informasi tentang pembaruan terbaru ke kumpulan data, Anda dapat menggunakan DescribeDataset operasi ini.

Mengekspor data pelatihan dalam kumpulan data ke Amazon S3

Setelah mengimpor data ke kumpulan data Amazon Personalize, Anda dapat mengekspor data ke bucket Amazon S3. Anda dapat mengekspor data untuk memverifikasi dan memeriksa data yang digunakan Amazon Personalize untuk menghasilkan rekomendasi, melihat peristiwa interaksi item yang sebelumnya Anda rekam secara real time, atau melakukan analisis offline pada data Anda.

Anda dapat memilih untuk mengekspor hanya data yang Anda impor secara massal (diimpor menggunakan pekerjaan impor kumpulan data Amazon Personalize), hanya data yang Anda impor satu per satu (catatan yang diimpor menggunakan konsol atau`PutEvents`,, atau `PutItems` operasi)`PutUsers`, atau keduanya.

> **Note**
>
> Anda tidak dapat mengekspor data dalam kumpulan data interaksi Tindakan atau kumpulan data Tindakan.

Untuk catatan yang sama persis dengan semua bidang, Amazon Personalize mengekspor hanya satu catatan. Jika dua rekaman memiliki ID yang sama tetapi satu atau beberapa bidang berbeda, Amazon Personalize menyertakan atau menghapus catatan tergantung pada data yang Anda pilih untuk diekspor:

- Jika Anda mengekspor data massal dan inkremental, Amazon Personalize hanya mengekspor item terbaru dengan ID yang sama (dalam ekspor kumpulan data Item), dan hanya pengguna dengan ID yang sama (dalam ekspor kumpulan data Pengguna). Untuk kumpulan data interaksi Item, Amazon Personalize mengekspor semua data interaksi item.
- Jika Anda hanya mengekspor data tambahan, Amazon Personalize mengekspor semua data interaksi item, pengguna, atau item yang Anda impor satu per satu, termasuk item atau pengguna yang sama. IDs Hanya catatan yang sama persis untuk semua bidang yang dikecualikan.
- Jika Anda hanya mengekspor data massal, Amazon Personalize menyertakan semua data interaksi item, pengguna, atau item yang Anda impor secara massal, termasuk item atau pengguna yang sama. IDs Hanya catatan yang sama persis untuk semua bidang yang dikecualikan.

Untuk mengekspor kumpulan data, Anda membuat pekerjaan ekspor kumpulan data. Pekerjaan ekspor kumpulan data adalah alat ekspor rekaman yang mengeluarkan catatan dalam kumpulan data ke satu atau beberapa file CSV di bucket Amazon S3. File CSV keluaran menyertakan baris header dengan nama kolom yang cocok dengan bidang dalam skema kumpulan data.

Topik

- Persyaratan izin pekerjaan ekspor dataset
- Membuat pekerjaan ekspor kumpulan data di Amazon Personalize

Persyaratan izin pekerjaan ekspor dataset

Untuk mengekspor kumpulan data, Amazon Personalize memerlukan izin untuk menambahkan file ke bucket Amazon S3 Anda. Untuk memberikan izin, lampirkan kebijakan baru AWS Identity and Access Management (IAM) ke peran layanan Amazon Personalisasi yang memberikan izin peran untuk menggunakan `PutObject` `ListBucket` dan Tindakan di bucket Anda, dan lampirkan kebijakan bucket ke bucket Amazon S3 keluaran Anda yang memberikan izin prinsip Personalisasi Amazon untuk menggunakan dan Tindakan. `PutObject` `ListBucket`

Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.

Kebijakan peran layanan untuk mengekspor kumpulan data

Kebijakan contoh berikut memberikan izin peran layanan Amazon Personalize Anda untuk menggunakan dan Tindakan`PutObject`. `ListBucket` Ganti `amzn-s3-demo-bucket` dengan nama bucket keluaran Anda. Untuk informasi tentang melampirkan kebijakan ke peran layanan IAM, lihat. Melampirkan kebijakan Amazon S3 ke peran layanan Amazon Personalisasi

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Action": [
```

```
                "s3:PutObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Kebijakan bucket Amazon S3 untuk mengekspor kumpulan data

Contoh kebijakan berikut memberikan izin Amazon Personalize untuk menggunakan `PutObject` `ListBucket` dan Tindakan di bucket Amazon S3. Ganti `amzn-s3-demo-bucket` dengan nama bucket Anda. Untuk informasi tentang menambahkan kebijakan bucket Amazon S3 ke bucket, lihat Menambahkan kebijakan bucket menggunakan konsol Amazon S3 di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "personalize.amazonaws.com"
            },
            "Action": [
                "s3:PutObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Membuat pekerjaan ekspor kumpulan data di Amazon Personalize

Anda dapat membuat tugas ekspor kumpulan data dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau. AWS SDKs

Membuat pekerjaan ekspor kumpulan data (konsol)

Setelah mengimpor data ke dalam kumpulan data dan membuat bucket Amazon S3 keluaran, Anda dapat mengekspor data ke bucket untuk dianalisis. Untuk mengekspor kumpulan data menggunakan konsol Amazon Personalize, Anda membuat tugas ekspor kumpulan data. Untuk informasi tentang membuat bucket Amazon S3, lihat Membuat bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Sebelum mengekspor kumpulan data, pastikan peran layanan Amazon Personalize dapat mengakses dan menulis ke bucket Amazon S3 keluaran Anda. Lihat Persyaratan izin pekerjaan ekspor dataset.

Untuk membuat pekerjaan ekspor kumpulan data (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/ rumah.
2. Di panel navigasi, pilih Grup kumpulan data.
3. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
4. Di panel navigasi, pilih Datasets.
5. Pilih kumpulan data yang ingin Anda ekspor ke bucket Amazon S3.
6. Di pekerjaan ekspor Dataset, pilih Buat pekerjaan ekspor kumpulan data.
7. Di detail pekerjaan ekspor Dataset, untuk nama pekerjaan ekspor Dataset, masukkan nama untuk pekerjaan ekspor.
8. Untuk peran layanan IAM, pilih peran layanan Amazon Personalisasi yang Anda buat. Membuat peran IAM untuk Amazon Personalize
9. Untuk jalur keluaran data Amazon S3, masukkan bucket Amazon S3 tujuan. Gunakan sintaks berikut:

   ```
   s3://amzn-s3-demo-bucket/<folder path>
   ```

10. Jika Anda menggunakan AWS KMS untuk enkripsi, untuk ARN kunci KMS, masukkan Nama Sumber Daya Amazon (ARN) untuk kunci tersebut. AWS KMS

11. Untuk tipe data Ekspor, pilih tipe data yang akan diekspor berdasarkan cara Anda mengimpor data semula.

    - Pilih Massal untuk mengekspor hanya data yang Anda impor secara massal menggunakan pekerjaan impor kumpulan data.
    - Pilih Incremental untuk mengekspor hanya data yang Anda impor satu per satu menggunakan konsol atau`PutEvents`,`PutUsers`, atau `PutItems` operasi.
    - Pilih Keduanya untuk mengekspor semua data dalam kumpulan data.
12. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya
13. Pilih Buat pekerjaan ekspor kumpulan data.

    Pada halaman ikhtisar Dataset, di pekerjaan ekspor Dataset, pekerjaan terdaftar dengan status pekerjaan Ekspor. Pekerjaan ekspor kumpulan data selesai ketika statusnya AKTIF. Anda kemudian dapat mengunduh data dari bucket Amazon S3 keluaran. Untuk informasi tentang mengunduh objek dari bucket Amazon S3, lihat Mengunduh objek di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. .

Membuat pekerjaan ekspor dataset ()AWS CLI

Setelah mengimpor data ke dalam kumpulan data dan membuat bucket Amazon S3 keluaran, Anda dapat mengekspor kumpulan data ke bucket untuk dianalisis. Untuk mengekspor kumpulan data menggunakan AWS CLI, buat pekerjaan ekspor kumpulan data menggunakan perintah. `create-dataset-export-job` AWS CLI Untuk informasi tentang membuat bucket Amazon S3, lihat Membuat bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Sebelum mengekspor kumpulan data, pastikan peran layanan Amazon Personalize dapat mengakses dan menulis ke bucket Amazon S3 keluaran Anda. Lihat Persyaratan izin pekerjaan ekspor dataset.

Berikut ini adalah contoh perintah `create-dataset-export-job` AWS CLI . Beri nama pekerjaan, ganti `dataset arn` dengan Amazon Resource Name (ARN) dari kumpulan data yang ingin Anda ekspor, dan ganti dengan ARN `role ARN` dari peran layanan Amazon Personalize yang Anda buat. Membuat peran IAM untuk Amazon Personalize Dis`3DataDestination`, untuk`kmsKeyArn`, secara opsional berikan ARN untuk kunci AWS KMS Anda, dan untuk `path` menyediakan jalur ke bucket Amazon S3 keluaran Anda.

Untuk`ingestion-mode`, tentukan data yang akan diekspor dari opsi berikut:

- Tentukan `BULK` untuk mengekspor hanya data yang Anda impor secara massal menggunakan pekerjaan impor kumpulan data.
- Tentukan `PUT` untuk mengekspor hanya data yang Anda impor satu per satu menggunakan konsol atau`PutEvents`, PutUsers, atau `PutItems` operasi.
- Tentukan `ALL` untuk mengekspor semua data dalam kumpulan data.

Untuk informasi selengkapnya, lihat CreateDatasetExportJob.

```
aws personalize create-dataset-export-job \
  --job-name job name \
  --dataset-arn dataset ARN \
  --job-output "{\"s3DataDestination\":{\"kmsKeyArn\":\"kms key ARN\",\"path\":
\"s3://amzn-s3-demo-bucket/folder-name/\"}}" \
  --role-arn role ARN \
  --ingestion-mode PUT
```

Pekerjaan ekspor dataset ARN ditampilkan.

```
{
  "datasetExportJobArn": "arn:aws:personalize:us-west-2:acct-id:dataset-export-job/
DatasetExportJobName"
}
```

Gunakan `DescribeDatasetExportJob` operasi untuk memeriksa status.

```
aws personalize describe-dataset-export-job \
  --dataset-export-job-arn dataset export job ARN
```

Membuat pekerjaan ekspor dataset ()AWS SDKs

Setelah mengimpor data ke dalam kumpulan data dan membuat bucket Amazon S3 keluaran, Anda dapat mengekspor kumpulan data ke bucket untuk dianalisis. Untuk mengekspor kumpulan data menggunakan AWS SDKs, buat pekerjaan ekspor kumpulan data menggunakan operasi. CreateDatasetExportJob Untuk informasi tentang membuat bucket Amazon S3, lihat Membuat bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Kode berikut menunjukkan cara membuat pekerjaan ekspor dataset menggunakan SDK for Python (Boto3) atau SDK for Java 2.x SDK.

Sebelum mengekspor kumpulan data, pastikan peran layanan Amazon Personalize dapat mengakses dan menulis ke bucket Amazon S3 keluaran Anda. Lihat Persyaratan izin pekerjaan ekspor dataset.

SDK for Python (Boto3)

Gunakan yang berikut ini `create_dataset_export_job` untuk mengekspor data dalam kumpulan data ke bucket Amazon S3. Beri nama pekerjaan, ganti `dataset arn` dengan Amazon Resource Name (ARN) dari kumpulan data yang ingin Anda ekspor, dan ganti dengan ARN `role` ARN dari peran layanan Amazon Personalize yang Anda buat. Membuat peran IAM untuk Amazon Personalize Dis`3DataDestination`, untuk`kmsKeyArn`, secara opsional berikan ARN untuk kunci AWS KMS Anda, dan untuk `path` menyediakan jalur ke bucket Amazon S3 keluaran Anda.

Untuk`ingestionMode`, tentukan data yang akan diekspor dari opsi berikut:

- Tentukan BULK untuk mengekspor hanya data yang Anda impor secara massal menggunakan pekerjaan impor kumpulan data.
- Tentukan PUT untuk mengekspor hanya data yang Anda impor satu per satu menggunakan konsol atau`PutEvents`, PutUsers, atau `PutItems` operasi.
- Tentukan ALL untuk mengekspor semua data dalam kumpulan data.

```
import boto3

personalize = boto3.client('personalize')

response = personalize.create_dataset_export_job(
    jobName = 'job name',
    datasetArn = 'dataset ARN',
    jobOutput = {
      "s3DataDestination": {
        "kmsKeyArn": "kms key ARN",
        "path": "s3://amzn-s3-demo-bucket/folder-name/"
      }
    },
    roleArn = 'role ARN',
    ingestionMode = 'PUT'
)
```

```
dsej_arn = response['datasetExportJobArn']

print ('Dataset Export Job arn: ' + dsej_arn)

description = personalize.describe_dataset_export_job(
    datasetExportJobArn = dsej_arn)['datasetExportJob']

print('Name: ' + description['jobName'])
print('ARN: ' + description['datasetExportJobArn'])
print('Status: ' + description['status'])
```

SDK for Java 2.x

Gunakan `createDatasetExportJob` metode berikut untuk membuat pekerjaan ekspor dataset. Teruskan yang berikut ini sebagai parameter: a PersonalizeClient, nama untuk pekerjaan ekspor Anda, ARN dari kumpulan data yang ingin Anda ekspor, mode konsumsi, jalur untuk bucket Amazon S3 keluaran, dan ARN untuk kunci Anda. AWS KMS

Ini `ingestionMode` bisa menjadi salah satu opsi berikut:

- Gunakan `IngestionMode.BULK` untuk mengekspor hanya data yang Anda impor secara massal menggunakan pekerjaan impor kumpulan data.
- Gunakan `IngestionMode.PUT` untuk mengekspor hanya data yang Anda impor satu per satu menggunakan konsol atau`PutEvents`, PutUsers, atau `PutItems` operasi.
- Gunakan `IngestionMode.ALL` untuk mengekspor semua data dalam dataset.

```
public static void createDatasetExportJob(PersonalizeClient personalizeClient,
                                           String jobName,
                                           String datasetArn,
                                           IngestionMode ingestionMode,
                                           String roleArn,
                                           String s3BucketPath,
                                           String kmsKeyArn) {

    long waitInMilliseconds = 30 * 1000; // 30 seconds
    String status = null;

    try {
        S3DataConfig exportS3DataConfig = S3DataConfig.builder()
            .path(s3BucketPath)
```

```
            .kmsKeyArn(kmsKeyArn)
            .build();

        DatasetExportJobOutput jobOutput = DatasetExportJobOutput.builder()
            .s3DataDestination(exportS3DataConfig)
            .build();

        CreateDatasetExportJobRequest createRequest =
CreateDatasetExportJobRequest.builder()
            .jobName(jobName)
            .datasetArn(datasetArn)
            .ingestionMode(ingestionMode)
            .jobOutput(jobOutput)
            .roleArn(roleArn)
            .build();

        String datasetExportJobArn =
personalizeClient.createDatasetExportJob(createRequest).datasetExportJobArn();

        DescribeDatasetExportJobRequest describeDatasetExportJobRequest =
DescribeDatasetExportJobRequest.builder()
            .datasetExportJobArn(datasetExportJobArn)
            .build();

        long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

        while (Instant.now().getEpochSecond() < maxTime) {

            DatasetExportJob datasetExportJob =
personalizeClient.describeDatasetExportJob(describeDatasetExportJobRequest)
                .datasetExportJob();

            status = datasetExportJob.status();
            System.out.println("Export job status: " + status);

            if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
                break;
            }
            try {
                Thread.sleep(waitInMilliseconds);
            } catch (InterruptedException e) {
                System.out.println(e.getMessage());
            }
        }
```

```
    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
}
```

Persyaratan untuk menghapus sumber daya Amazon Personalize

Menghapus sumber daya dapat membantu Anda menghindari biaya yang tidak perlu. Misalnya, Anda dikenakan biaya kampanye saat kampanye aktif. Untuk menghindari biaya yang tidak perlu, pastikan untuk menghapus kampanye ketika Anda selesai. Untuk daftar lengkap biaya dan harga, lihat Harga Amazon Personalisasi.

Untuk menghapus sumber daya dengan konsol Amazon Personalize, Anda memilih Hapus pada halaman detail untuk sumber daya. Untuk menghapus sumber daya dengan Amazon Personalize APIs, Anda menggunakan `Delete` APIs with SDKs atau AWS Command Line Interface ()AWS CLI.

Untuk langkah-langkah mendetail untuk menghapus kumpulan data dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau, AWS SDKs lihat. Menghapus dataset untuk menghapus semua datanya Anda dapat menerapkan pola dalam langkah-langkah ini ke sumber daya Amazon Personalize lainnya. Untuk informasi tentang menghapus pengguna dan datanya dari grup kumpulan data Anda, lihat. Menghapus pengguna dan data mereka dengan pekerjaan penghapusan data

Anda harus menghapus beberapa sumber daya sebelum Anda dapat menghapus yang lain. Misalnya, jika Anda membuat pelacak peristiwa dan kumpulan data interaksi Item, Anda harus menghapus pelacak peristiwa sebelum dapat menghapus kumpulan data. Bagian berikut menyediakan pedoman dan persyaratan pesanan untuk menghapus sumber daya Amazon Personalize.

Topik

- Pedoman untuk menghapus sumber daya
- Pesanan yang disarankan untuk penghapusan sumber daya
- Menghapus pengguna dan data mereka dengan pekerjaan penghapusan data
- Menghapus dataset untuk menghapus semua datanya

Pedoman untuk menghapus sumber daya

Berikut ini adalah pedoman untuk menghapus sumber daya:

- Menghapus sumber daya di Amazon Personalize adalah tindakan yang tidak dapat diubah. Penghapusan tidak dapat dihentikan setelah dimulai.
- Anda tidak dapat menghapus sumber daya yang statusnya berubah dari satu status ke status lainnya. Misalnya, Anda tidak dapat menghapus sumber daya yang CREATE PENDING atau IN PROGRESS. Status sumber daya harus AKTIF atau CREATE FAILED. Ini termasuk `latestSolutionUpdate` status untuk solusi. Anda dapat memeriksa status sumber daya menggunakan file `Describe` APIs. Misalnya, operasi DescribeCampaign API.
- Untuk informasi tentang menghapus data pelatihan di Amazon S3, lihat Bagaimana cara menghapus objek dari bucket S3? .
- Anda tidak dikenakan biaya untuk pekerjaan impor dataset setelah selesai dan Anda tidak dapat menghapusnya.
- Anda tidak dikenakan biaya untuk skema dan Anda tidak dapat menghapus skema dengan konsol Amazon Personalize. Untuk menghapus skema, gunakan operasi DeleteSchema API.

Berikut ini adalah persyaratan khusus untuk menghapus kumpulan data:

- Anda harus menghapus semua filter sebelum menghapus kumpulan data apa pun.
- Jika Anda membuat pelacak peristiwa, Anda harus menghapusnya sebelum menghapus kumpulan data interaksi Item.
- Jika Anda membuat atribusi metrik yang mereferensikan kumpulan data, Anda harus menghapus atribusi metrik terlebih dahulu.
- Jika Anda menggunakan User-Personalization-v 2, Personalisasi Pengguna, atau Next-Best-Action resep atau Pilihan teratas untuk Anda dan Direkomendasikan untuk kasus penggunaan Anda, menghapus kumpulan data akan menghentikan pembaruan otomatis untuk versi atau pemberi rekomendasi solusi terkait.
- Tidak ada yang terkait `DatasetImportJob` dapat memiliki status CREATE PENDING atau IN PROGRESS.
- Tidak ada yang terkait `BatchInferenceJob` atau `BatchSegmentJob` dapat memiliki status CREATE PENDING atau IN PROGRESS.
- Tidak terkait`Recommender`, `SolutionVersion` dapat memiliki status CREATE PENDING atau IN PROGRESS.
- Tidak ada yang terkait `Campaign` dapat memiliki status CREATE PENDING atau IN PROGRESS atau ACTIVE.

Pesanan yang disarankan untuk penghapusan sumber daya

Untuk menghindari kesalahan penghapusan, sebaiknya Anda menghapus sumber daya dari grup kumpulan data dalam urutan berikut. Untuk mengidentifikasi sumber daya dalam grup kumpulan data, Anda dapat menggunakan operasi API Daftar. Misalnya, Anda dapat menggunakan operasi ListFilters API untuk mengidentifikasi semua filter dalam grup kumpulan data.

1. Kampanye atau pemberi rekomendasi apa pun — Untuk menghapus kampanye atau pemberi rekomendasi Anda dengan APIs, gunakan operasi DeleteCampaign atau DeleteRecommender API. Dengan pemberi rekomendasi, Anda dapat menghentikan pemberi rekomendasi dan memulainya nanti. Dengan cara ini, Anda dapat menjeda penagihan pemberi rekomendasi dan hanya membayarnya saat Anda menggunakannya. Untuk informasi selengkapnya, lihat Menghentikan pemberi rekomendasi.
2. Solusi apa pun — Untuk menghapus solusi Anda dengan APIs, gunakan operasi DeleteSolution API. Untuk menghapus solusi, pembaruan solusi tidak dapat dilakukan. `latestSolutionUpdate`Statusnya harus AKTIF atau CREATE FAILED. Menghapus solusi akan menghapus semua versi solusi terkait. Tak satu pun dari versi solusinya dapat memiliki status CREATE PENDING atau IN PROGRESS.
3. Pelacak peristiwa — Untuk menghapus pelacak peristiwa dengan APIs, gunakan operasi DeleteEventTracker API. Anda harus menghapus pelacak acara sebelum dapat menghapus kumpulan data interaksi Item.
4. Atribusi metrik — Untuk menghapus atribusi metrik dengan APIs, gunakan operasi DeleteMetricAttribution API.
5. Semua filter — Untuk menghapus filter dengan APIs, gunakan operasi DeleteFilter API. Anda harus menghapus semua filter sebelum menghapus kumpulan data.
6. Kumpulan data apa pun — Untuk menghapus kumpulan data dengan APIs, gunakan operasi API. DeleteDataset
7. Grup kumpulan data — Untuk menghapus grup kumpulan data Anda dengan APIs, gunakan operasi DeleteDatasetGroup API.
8. Skema — Untuk menghapus skema, gunakan operasi DeleteSchema API.

Menghapus pengguna dan data mereka dengan pekerjaan penghapusan data

Setelah mengimpor data, Anda dapat menghapus pengguna dan datanya, termasuk metadata dan data interaksinya, dari grup kumpulan data. Anda dapat menghapus data pengguna sebagai bagian dari program kepatuhan, atau untuk mengatasi permintaan penghapusan pengguna, atau untuk menjaga data Anda tetap terkini saat basis pengguna Anda berubah.

Setelah Anda menghapus pengguna, Amazon Personalize tidak lagi melatih data mereka dan tidak lagi mempertimbangkan pengguna saat membuat segmen pengguna.

Untuk menghapus referensi ke pengguna di Amazon Personalisasi kumpulan data dan model dalam grup kumpulan data, Anda melakukan hal berikut:

1. Siapkan file CSV yang mencantumkan userIds pengguna yang akan dihapus di kolom USER_ID.
2. Unggah file CSV ke bucket Amazon S3. Peran layanan Personalisasi Amazon Anda harus memiliki izin untuk mengakses bucket ini.
3. Buat pekerjaan penghapusan data. Pekerjaan penghapusan data adalah pekerjaan batch yang menghapus pengguna dan data mereka dari model dan kumpulan data dalam grup dataset.

Topik

- Pedoman dan persyaratan
- Mempersiapkan daftar pengguna untuk dihapus
- Membuat pekerjaan penghapusan data

Pedoman dan persyaratan

Berikut ini adalah pedoman dan persyaratan untuk menghapus pengguna:

- Sebelum Anda membuat pekerjaan penghapusan data, pastikan tidak ada pekerjaan yang menggunakan kumpulan data Anda yang sedang berlangsung, seperti pekerjaan pelatihan, pekerjaan batch, atau operasi impor massal atau individual. Dan hindari membuat pekerjaan seperti itu saat pekerjaan penghapusan data sedang berlangsung. Jika ada pelatihan atau impor yang terjadi, kami tidak dapat menjamin bahwa data pengguna akan dihapus dari model dan kami sarankan untuk membuat pekerjaan penghapusan data tambahan.

- Pekerjaan penghapusan data tidak menghapus referensi ke pengguna di luar Amazon Personalize. Misalnya, UserID tidak dihapus dari rekomendasi batch di bucket Amazon S3 Anda. Anda harus menghapus catatan ini secara manual.
- Anda dapat memiliki hingga 5 pekerjaan penghapusan untuk grup kumpulan data dengan status PENDING.
- Ukuran total maksimum file input penghapusan data atau file Anda adalah 100 MB. Anda dapat menggunakan kembali file input yang sama saat Anda membuat pekerjaan penghapusan.
- Setiap pekerjaan penghapusan data menghapus pengguna dan data interaksi mereka dalam grup kumpulan data. Untuk menghapus datanya di semua grup kumpulan data, Anda harus membuat pekerjaan penghapusan data untuk setiap grup kumpulan data.
- Setelah Anda membuat pekerjaan, diperlukan waktu hingga satu hari untuk menghapus data pengguna dari kumpulan data dan model.
- Setelah pekerjaan selesai, pastikan untuk memperbarui sumber daya khusus apa pun. Pastikan untuk membuat versi solusi baru dan, jika perlu, perbarui kampanye Anda. Jika Anda menggunakan pelatihan otomatis, Anda masih dapat membuat versi solusi baru secara manual.
- Peran layanan Personalisasi Amazon Anda harus memiliki izin untuk mengakses bucket Amazon S3 Anda dengan daftar pengguna yang akan dihapus. Kebutuhan `GetObject` dan `ListBucket` izin untuk ember dan kontennya. Izin ini sama dengan mengimpor data. Untuk informasi tentang pemberian izin dan contoh kebijakan, lihat. Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3
- Anda tidak dapat menggunakan AWS Key Management Service kunci Anda sendiri di bucket Amazon S3 yang menyimpan daftar UserIds pengguna untuk dihapus.
- Jika item hanya muncul di kumpulan data kumpulan data interaksi Item Anda dan hanya pengguna yang Anda hapus yang berinteraksi dengan item ini, item ini tidak akan lagi muncul di rekomendasi.

Mempersiapkan daftar pengguna untuk dihapus

Sebelum menghapus pengguna dari Amazon Personalize, Anda harus menyiapkan daftar pengguna untuk dihapus dalam file CSV dan mengunggahnya ke Amazon S3.

Untuk mempersiapkan daftar pengguna untuk menghapus dan mengunggahnya

1. Buat file CSV yang mencantumkan userIds pengguna yang akan dihapus. Berikut ini menunjukkan bagaimana file CSV Anda harus diformat.

```
USER_ID
abc
2a
5basc
ab35
123f
a55d
0v22
441fa
efg
```

2. Unggah file CSV Anda ke bucket Amazon Simple Storage Service (Amazon S3). Untuk informasi selengkapnya tentang mengunggah file ke Amazon S3, lihat Mengunggah File dan Folder dengan Menggunakan Seret dan Jatuhkan di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.
3. Berikan Amazon Personalize akses ke bucket dan file CSV Anda. Amazon Personalize harus memiliki izin untuk melakukan `GetObject` dan `ListBucket` Tindakan pada bucket Anda dan isinya. Izin ini sama dengan mengimpor data. Untuk informasi tentang pemberian izin dan contoh kebijakan, lihat. Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3

Membuat pekerjaan penghapusan data

Setelah selesaiMempersiapkan daftar pengguna untuk dihapus, Anda siap untuk menghapus pengguna dengan pekerjaan penghapusan data.

Pekerjaan penghapusan data adalah pekerjaan batch yang menghapus pengguna dan data mereka dari model dan kumpulan data dalam grup dataset. Setelah Anda menghapus pengguna, Amazon Personalize tidak lagi melatih data mereka dan tidak lagi mempertimbangkan pengguna saat membuat segmen pengguna.

Saat membuat pekerjaan penghapusan data, Anda menentukan lokasi Amazon S3 dari daftar pengguna yang akan dihapus.

- Jika data Anda dalam satu file, gunakan sintaks berikut untuk lokasi Amazon S3:

  `s3://amzn-s3-demo-bucket/<folder path>/<CSV filename>.csv`
- Jika file CSV Anda berada di folder di bucket Amazon S3, Anda dapat menentukan jalur ke folder tersebut. Dengan pekerjaan penghapusan data, Amazon Personalize menggunakan semua file

dengan ekstensi file `.csv` di folder dan sub folder apa pun. Ini mengabaikan file dari jenis lain. Gunakan sintaks berikut dengan nama folder `/` setelah:

`s3://amzn-s3-demo-bucket/<folder path>/`

Peran yang Anda gunakan harus memiliki izin untuk melakukan `GetObject` dan `ListBucket` Tindakan di bucket Amazon S3 Anda dan isinya. Untuk informasi tentang pemberian izin dan contoh kebijakan, lihat. Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3

Anda dapat membuat tugas penghapusan data dengan konsol Amazon Personalize,AWS CLI(), AWS Command Line Interface atau. AWS SDKs

Membuat pekerjaan penghapusan data (konsol)

Untuk menghapus pengguna dengan konsol Amazon Personalize, buat tugas penghapusan data dengan nama, peran layanan IAM, dan lokasi Amazon S3 data Anda.

Untuk menghapus catatan (konsol)

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pada halaman grup Dataset, pilih grup kumpulan data Anda. Ikhtisar grup kumpulan data ditampilkan.
3. Di panel navigasi, pilih Datasets.
4. Dalam pekerjaan penghapusan data, pilih Buat pekerjaan.
5. Dalam rincian Job, beri nama pekerjaan itu.
6. Di sumber Input S3, untuk Lokasi S3, tentukan lokasi Amazon S3 dari file CSV yang menyimpan daftar userIds pengguna yang akan dihapus. Anda menyiapkan file ini diMempersiapkan daftar pengguna untuk dihapus.
7. Dalam peran IAM, pilih untuk membuat peran baru atau menggunakan yang sudah ada. Jika Anda menyelesaikan prasyarat untuk membuat peran untuk Amazon Personalisasi dan memberikan akses peran ini ke bucket Amazon S3, pilih Gunakan peran layanan yang ada dan tentukan peran yang Anda buat. Membuat peran IAM untuk Amazon Personalize

   Peran yang Anda gunakan harus memiliki izin untuk melakukan `GetObject` dan `ListBucket` Tindakan di bucket Amazon S3 Anda dan isinya. Izin ini sama dengan mengimpor data. Untuk informasi tentang pemberian izin dan contoh kebijakan, lihat. Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3

8. Untuk Tag, secara opsional tambahkan tag apa pun. Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya
9. Pilih Buat tugas. Pekerjaan dimulai dan halaman detail ditampilkan.

   Setelah Anda membuat pekerjaan, itu bisa sekitar satu hari untuk menghapus data pengguna dari kumpulan data dan model. Sampai pekerjaan selesai, Amazon Personalize terus menggunakan data saat pelatihan. Dan pengguna mungkin muncul di segmen pengguna.

   Penghapusan data selesai ketika status ditampilkan sebagai SELESAI. Jika pekerjaan gagal karena alasan apa pun, sebaiknya buat pekerjaan penghapusan data lainnya. Setelah pekerjaan selesai, pastikan untuk memperbarui sumber daya khusus apa pun. Pastikan untuk membuat versi solusi baru dan, jika perlu, perbarui kampanye Anda. Jika Anda menggunakan pelatihan otomatis, Anda masih dapat membuat versi solusi baru secara manual.

Membuat pekerjaan penghapusan data ()AWS CLI

Untuk menghapus pengguna dengan AWS CLI, gunakan `create-data-deletion-job` perintah. Perintah ini menggunakan operasi CreateDataDeletion API. Kode berikut menunjukkan cara membuat pekerjaan penghapusan data. Untuk menggunakan kode, perbarui kode tersebut untuk menentukan nama pekerjaan, peran IAM yang Anda buatMembuat peran IAM untuk Amazon Personalize, dan lokasi Amazon S3 data Anda. Anda menyiapkan file ini diMempersiapkan daftar pengguna untuk dihapus.

```
aws personalize create-data-deletion-job \
--job-name deletion job name \
--dataset-group-arn dataset group ARN \
--data-source dataLocation=s3://amzn-s3-demo-bucket/filename.csv \
--role-arn roleArn
```

Setelah Anda membuat pekerjaan, itu bisa sekitar satu hari untuk menghapus data pengguna dari kumpulan data dan model. Sampai pekerjaan selesai, Amazon Personalize terus menggunakan data saat pelatihan. Dan pengguna mungkin muncul di segmen pengguna.

Pekerjaan selesai ketika status selesai. Periksa status dengan menggunakan `describe-data-deletion-job` perintah dan tentukan pekerjaan penghapusan data ARN. Untuk informasi selengkapnya tentang operasi API, lihatDescribeDataDeletionJob. Untuk melihat riwayat pekerjaan

penghapusan data yang diurutkan berdasarkan waktu pembuatan, gunakan operasi API. ListDataDeletionJobs

Jika pekerjaan gagal karena alasan apa pun, sebaiknya buat pekerjaan penghapusan data lainnya. Setelah pekerjaan selesai, pastikan untuk memperbarui sumber daya khusus apa pun. Pastikan untuk membuat versi solusi baru dan, jika perlu, perbarui kampanye Anda. Jika Anda menggunakan pelatihan otomatis, Anda masih dapat membuat versi solusi baru secara manual.

Membuat pekerjaan penghapusan data ()AWS SDKs

Untuk menghapus pengguna dengan AWS SDKs, gunakan operasi CreateDataDeletionJob API. Kode berikut menunjukkan cara membuat pekerjaan penghapusan data. Untuk menggunakan kode, perbarui kode tersebut untuk menentukan nama pekerjaan, peran IAM yang Anda buatMembuat peran IAM untuk Amazon Personalize, dan lokasi Amazon S3 data Anda. Anda menyiapkan file ini diMempersiapkan daftar pengguna untuk dihapus.

```
import boto3

personalize = boto3.client('personalize')

response = personalize.create_data_deletion_job(
    jobName = 'Deletion job name',
    datasetGroupArn = 'Dataset Group ARN',
    dataSource = {'dataLocation':'s3://amzn-s3-demo-bucket/file.csv'},
    roleArn = 'role_arn'
)

deletion_job_arn = response['dataDeletionJobArn']

print ('Deletion Job arn: ' + deletion_job_arn)

description = personalize.describe_data_deletion_job(
    dataDeletionJobArn = deletion_job_arn)['dataDeletionJob']

print('Name: ' + description['jobName'])
print('ARN: ' + description['dataDeletionJobArn'])
print('Status: ' + description['status'])
```

Setelah Anda membuat pekerjaan, itu bisa sekitar satu hari untuk menghapus data pengguna dari kumpulan data dan model. Sampai pekerjaan selesai, Amazon Personalize terus menggunakan data saat pelatihan. Dan pengguna mungkin muncul di segmen pengguna.

Pekerjaan selesai ketika status selesai. Periksa status dengan menggunakan DescribeDataDeletionJob operasi dan tentukan pekerjaan penghapusan data ARN. Untuk melihat riwayat pekerjaan penghapusan data yang diurutkan berdasarkan waktu pembuatan, gunakan operasi API. ListDataDeletionJobs

Jika pekerjaan gagal karena alasan apa pun, sebaiknya buat pekerjaan penghapusan data lainnya. Setelah pekerjaan selesai, pastikan untuk memperbarui sumber daya khusus apa pun. Pastikan untuk membuat versi solusi baru dan, jika perlu, perbarui kampanye Anda. Jika Anda menggunakan pelatihan otomatis, Anda masih dapat membuat versi solusi baru secara manual.

Menghapus dataset untuk menghapus semua datanya

Untuk menghapus semua data dalam kumpulan data, Anda menghapus kumpulan data. Anda dapat menghapus kumpulan data dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau. AWS SDKs Sebelum Anda menghapus kumpulan data, perhatikan hal berikut:

- Anda harus menghapus semua filter sebelum menghapus kumpulan data apa pun.
- Jika Anda membuat pelacak peristiwa, Anda harus menghapusnya sebelum menghapus kumpulan data interaksi Item.
- Jika Anda membuat atribusi metrik yang mereferensikan kumpulan data, Anda harus menghapus atribusi metrik terlebih dahulu.
- Jika Anda menggunakan User-Personalization-v 2, Personalisasi Pengguna, atau Next-Best-Action resep atau Pilihan teratas untuk Anda dan Direkomendasikan untuk kasus penggunaan Anda, menghapus kumpulan data akan menghentikan pembaruan otomatis untuk versi atau pemberi rekomendasi solusi terkait.
- Tidak ada yang terkait `DatasetImportJob` dapat memiliki status CREATE PENDING atau IN PROGRESS.
- Tidak ada yang terkait `BatchInferenceJob` atau `BatchSegmentJob` dapat memiliki status CREATE PENDING atau IN PROGRESS.
- Tidak ada yang terkait`Recommender`, `SolutionVersion` dapat memiliki status CREATE PENDING atau IN PROGRESS.
- Tidak ada yang terkait `Campaign` dapat memiliki status CREATE PENDING atau IN PROGRESS atau ACTIVE.

Topik

- Menghapus dataset (konsol)
- Menghapus dataset ()AWS CLI
- Menghapus dataset ()AWS SDKs

Menghapus dataset (konsol)

Untuk menghapus kumpulan data dengan konsol Amazon Personalize, navigasikan ke halaman detail kumpulan data dan pilih hapus.

Untuk menghapus dataset

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/ rumah.
2. Di panel navigasi, pilih Grup kumpulan data.
3. Pada halaman grup Dataset, pilih grup kumpulan data Anda.
4. Di panel navigasi, pilih Datasets.
5. Pilih kumpulan data untuk membuka halaman detailnya.
6. Pada halaman detail kumpulan data, pilih Hapus dan konfirmasikan penghapusan kumpulan data.

Menghapus dataset ()AWS CLI

Kode berikut menunjukkan cara menghapus dataset dengan AWS CLI dan DeleteDataset operasi.

```
aws personalize delete-dataset --dataset-arn dataset-arn
```

Menghapus dataset ()AWS SDKs

Kode berikut menunjukkan cara menghapus dataset dengan AWS SDKs dan DeleteDataset operasi.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.delete_dataset(
    datasetArn = 'dataset ARN'
```

```
)
```

SDK for Java 2.x

```
public static void deleteDataset(PersonalizeClient personalizeClient,
                                 String datasetArn) {

    try {
        DeleteDatasetRequest deleteRequest = DeleteDatasetRequest.builder()
                .datasetArn(datasetArn)
                .build();

        int responseCode =
 personalizeClient.deleteDataset(deleteRequest).sdkHttpResponse().statusCode();
        System.out.println(responseCode);
    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
}
```

Mengukur dampak rekomendasi Amazon Personalize

Saat pelanggan berinteraksi dengan rekomendasi, Anda dapat mengukur bagaimana rekomendasi Amazon Personalisasi membantu Anda mencapai tujuan Anda. Anda dapat mengidentifikasi kampanye dan pemberi rekomendasi mana yang paling berdampak pada metrik kinerja utama. Misalnya, Anda dapat mengidentifikasi sumber daya mana yang menghasilkan menit paling banyak ditonton, klik terbanyak, atau pembelian terbanyak. Dan Anda dapat membandingkan kinerja rekomendasi Amazon Personalize dengan yang dihasilkan oleh layanan pihak ketiga.

Ketika Anda tahu kampanye atau pemberi rekomendasi mana yang menghasilkan dampak paling besar, Anda dapat mengambil tindakan untuk mendapatkan manfaat lebih lanjut dari rekomendasinya. Misalnya, Anda dapat meningkatkan keunggulan rekomendasi di situs Anda untuk mendorong lebih banyak keterlibatan. Atau Anda mungkin menampilkan rekomendasi dalam kampanye pemasaran, seperti email yang dipersonalisasi atau iklan bertarget.

Jika Anda mengidentifikasi sumber daya yang tidak memiliki dampak yang diharapkan, Anda dapat mengambil tindakan untuk meningkatkan rekomendasi. Misalnya, Anda dapat menggunakan konsol Amazon Personalize untuk menganalisis data pelatihan yang digunakan untuk membuat sumber daya, melakukan peningkatan data yang disarankan, dan kemudian mengimpor data lagi. Untuk informasi selengkapnya tentang menganalisis data, lihatMenganalisis kualitas dan kuantitas data di Amazon Personalisasi kumpulan data.

Berikut ini dapat membantu Anda mengukur dampak rekomendasi:

- Atribusi metrik: Atribusi metrik Personalisasi Amazon membuat laporan berdasarkan metrik yang Anda tentukan serta interaksi item serta data item yang Anda impor. Misalnya, total panjang film yang ditonton oleh pengguna, atau jumlah total peristiwa klik. Setelah Anda membuat atribusi metrik, Amazon Personalize secara otomatis mengirimkan metrik pada peristiwa dari PutEvents operasi API dan data massal tambahan ke Amazon. CloudWatch Untuk data massal, Anda dapat memilih untuk mempublikasikan laporan ke bucket Amazon S3.
- Pengujian A/B: Melakukan A/B pengujian dengan rekomendasi Amazon Personalize melibatkan menampilkan berbagai kelompok pengguna berbagai jenis rekomendasi dan membandingkan hasil. Anda dapat menggunakan A/B pengujian untuk membantu membandingkan dan mengevaluasi berbagai strategi rekomendasi, mengevaluasi kinerja model, dan mengukur dampak rekomendasi.

Topik

- Mengukur dampak rekomendasi dengan atribusi metrik
- Mengukur dampak rekomendasi dengan A/B pengujian

Mengukur dampak rekomendasi dengan atribusi metrik

Untuk mengukur dampak rekomendasi item, Anda dapat membuat atribusi metrik. Atribusi metrik membuat laporan berdasarkan interaksi item dan data item yang Anda impor, dan metrik yang Anda tentukan. Misalnya, total panjang film yang ditonton oleh pengguna, atau jumlah total peristiwa klik. Amazon Personalisasi perhitungan agregat selama 15 menit. Untuk data interaksi streaming dan data massal tambahan, Amazon Personalize secara otomatis mengirimkan laporan metrik ke Amazon. CloudWatch Untuk data massal, Anda dapat memilih untuk mempublikasikan laporan ke bucket Amazon S3.

Untuk setiap interaksi yang Anda impor, sertakan data sumber untuk membandingkan kampanye, pemberi rekomendasi, dan pihak ketiga yang berbeda. Anda dapat menyertakan ID rekomendasi dari rekomendasi yang Anda tunjukkan kepada pengguna atau sumber acara, seperti pihak ketiga.

Misalnya, Anda mungkin memiliki aplikasi streaming video yang menampilkan rekomendasi film dari dua rekomendasi Amazon Personalize yang berbeda. Jika Anda ingin melihat pemberi rekomendasi mana yang menghasilkan acara tontonan terbanyak, Anda dapat membuat atribusi metrik yang melacak jumlah total acara tontonan. Kemudian Anda dapat merekam acara tontonan saat pengguna berinteraksi dengan rekomendasi, dan menyertakan `recommendationId` di setiap acara. Amazon Personalize menggunakan `recommendationId` untuk mengidentifikasi setiap pemberi rekomendasi. Saat merekam acara, Anda dapat melihat total acara tontonan yang dikumpulkan setiap 15 menit untuk kedua pemberi rekomendasi. CloudWatch Untuk contoh kode yang menunjukkan cara menyertakan `recommendationId` atau `eventAttributionSource` untuk suatu peristiwa, lihatMetrik peristiwa dan laporan atribusi.

Topik

- Pedoman dan persyaratan untuk atribusi metrik
- Membuat atribusi metrik Amazon Personalisasi
- Memperbarui atribusi metrik Amazon Personalisasi
- Menghapus atribusi metrik Amazon Personalisasi
- Melihat grafik data metrik di CloudWatch
- Menerbitkan laporan atribusi metrik ke Amazon S3

Pedoman dan persyaratan untuk atribusi metrik

Amazon Personalize mulai menghitung dan melaporkan dampak rekomendasi hanya setelah Anda membuat atribusi metrik. Untuk membuat riwayat terlengkap, sebaiknya buat atribusi metrik sebelum mengimpor data interaksi. Saat membuat tugas impor kumpulan data untuk kumpulan data interaksi Item dengan konsol Amazon Personalize, Anda memiliki opsi untuk membuat atribusi metrik di tab baru. Kemudian Anda dapat kembali ke pekerjaan impor untuk menyelesaikannya.

Setelah Anda membuat atribusi metrik dan merekam peristiwa atau mengimpor data massal tambahan, Anda akan dikenakan CloudWatch biaya bulanan per metrik. Untuk informasi tentang CloudWatch harga, lihat halaman CloudWatch harga Amazon. Untuk berhenti mengirim metrik ke CloudWatch, hapus atribusi metrik.

Untuk melihat dampak rekomendasi dari waktu ke waktu, teruslah mengimpor data saat pelanggan berinteraksi dengan rekomendasi. Jika Anda telah mengimpor data, Anda masih dapat membuat atribusi metrik dan mulai mengukur dampak rekomendasi. Namun, Amazon Personalize tidak akan melaporkan data yang diimpor sebelum membuatnya.

Berikut ini adalah pedoman dan persyaratan untuk menghasilkan laporan dengan atribusi metrik:

- Anda harus memberikan izin Amazon Personalize untuk mengakses dan memasukkan data. CloudWatch Untuk contoh kebijakan, lihatMemberikan Amazon Personalisasi akses ke CloudWatch.
- Untuk memublikasikan metrik ke Amazon S3, berikan izin Amazon Personalize untuk menulis ke bucket Anda. Anda juga harus menyediakan jalur bucket dalam atribusi metrik Anda. Untuk contoh kebijakan, lihatMemberikan Amazon Personalisasi akses ke bucket Amazon S3 Anda.
- Untuk mempublikasikan metrik ke CloudWatch, catatan harus berusia kurang dari 14 hari. Jika data Anda lebih tua, catatan ini tidak akan disertakan dalam perhitungan atau laporan.
- Mengimpor peristiwa duplikat (peristiwa yang sama persis dengan semua atribut) dapat menyebabkan perilaku tak terduga termasuk metrik yang tidak akurat. Kami menyarankan Anda menghapus catatan duplikat dari data massal apa pun sebelum mengimpor, dan menghindari mengimpor kejadian duplikat dengan operasi. `PutEvents`
- Dataset interaksi Item Anda harus memiliki `EVENT_TYPE` kolom.
- Anda tidak dapat membuat laporan metrik untuk data dalam kumpulan data interaksi Tindakan.
- Anda dapat membuat paling banyak satu atribusi metrik per grup kumpulan data. Setiap atribusi metrik dapat memiliki paling banyak 10 metrik.

Untuk membandingkan sumber, setiap peristiwa interaksi harus menyertakan `recommendationId` atau`eventAttributionSource`. Anda dapat menyediakan paling banyak 100 sumber atribusi acara unik. Untuk sampel kode `PutEvents`, lihat Metrik peristiwa dan laporan atribusi.

- Jika Anda memberikan`recommendationId`, Amazon Personalize secara otomatis menentukan kampanye sumber atau pemberi rekomendasi dan mengidentifikasinya dalam laporan di kolom EVENT_ATTRIBUTION_SOURCE.
- Jika Anda memberikan kedua atribut tersebut, Amazon Personalize hanya menggunakan fitur. `eventAttributionSource`
- Jika Anda tidak menyediakan sumber, Amazon Personalize memberi label pada sumber `SOURCE_NAME_UNDEFINED` dalam laporan.

Topik

- Memberikan Amazon Personalisasi akses ke CloudWatch
- Memberikan Amazon Personalisasi akses ke bucket Amazon S3 Anda

Memberikan Amazon Personalisasi akses ke CloudWatch

> **Important**
>
> Saat Anda memberikan izin, Amazon Personalisasi tempat dan memvalidasi sejumlah kecil data. CloudWatch Ini akan dikenakan biaya satu kali kurang dari $0,30. Untuk informasi selengkapnya tentang CloudWatch harga, lihat halaman CloudWatch harga Amazon.

Untuk memberikan akses Amazon Personalize CloudWatch, lampirkan kebijakan baru AWS Identity and Access Management (IAM) ke peran layanan Amazon Personalize yang memberikan izin peran untuk menggunakan Action for. `PutMetricData` CloudWatch Contoh kebijakan berikut memberikan `PutMetricData` izin.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
      "Effect": "Allow",
      "Action": [
        "cloudwatch:PutMetricData"
      ],
      "Resource": "*"
    }
  ]
}
```

Memberikan Amazon Personalisasi akses ke bucket Amazon S3 Anda

Untuk memberi Amazon Personalisasi akses ke bucket Amazon S3 Anda:

- Lampirkan kebijakan IAM ke peran layanan Amazon Personalize yang memberikan izin peran untuk menggunakan Tindakan `PutObject` di bucket Anda.

  JSON

  ```
  {
      "Version": "2012-10-17",
      "Id": "PersonalizeS3BucketAccessPolicy",
      "Statement": [
          {
              "Sid": "PersonalizeS3BucketAccessPolicy",
              "Effect": "Allow",
              "Action": [
                  "s3:PutObject"
              ],
              "Resource": [
                  "arn:aws:s3:::amzn-s3-demo-bucket",
                  "arn:aws:s3:::amzn-s3-demo-bucket/*"
              ]
          }
      ]
  }
  ```

- Lampirkan kebijakan bucket ke bucket Amazon S3 keluaran Anda yang memberikan izin prinsip Personalisasi Amazon untuk menggunakan Tindakan. `PutObject`

  Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk

menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.

JSON

```
{
    "Version": "2012-10-17",
    "Id": "PersonalizeS3BucketAccessPolicy",
    "Statement": [
        {
            "Sid": "PersonalizeS3BucketAccessPolicy",
            "Effect": "Allow",
            "Principal": {
                "Service": "personalize.amazonaws.com"
            },
            "Action": [
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Membuat atribusi metrik Amazon Personalisasi

> **Important**
>
> Setelah Anda membuat atribusi metrik dan merekam peristiwa atau mengimpor data massal tambahan, Anda akan dikenakan CloudWatch biaya bulanan per metrik. Untuk informasi tentang CloudWatch harga, lihat halaman CloudWatch harga Amazon. Untuk berhenti mengirim metrik ke CloudWatch, hapus atribusi metrik.

Untuk mulai membuat laporan metrik, Anda membuat atribusi metrik dan mengimpor data interaksi. Saat membuat atribusi metrik, Anda menentukan daftar jenis peristiwa yang akan dilaporkan. Untuk setiap jenis peristiwa, Anda menentukan fungsi yang diterapkan Amazon Personalize

saat mengumpulkan data. Fungsi yang tersedia termasuk `SUM(DatasetType.COLUMN_NAME)` dan`SAMPLECOUNT()`.

Misalnya, Anda mungkin memiliki aplikasi streaming video online dan ingin melacak dua metrik: rasio klik-tayang untuk rekomendasi, dan total panjang film yang ditonton, di mana setiap video dalam kumpulan data Item menyertakan atribut. `LENGTH` Anda akan membuat atribusi metrik dan menambahkan dua metrik, masing-masing dengan jenis dan fungsi peristiwa. Yang pertama mungkin untuk jenis `Click` acara dengan `SAMPLECOUNT()` fungsi. Yang kedua mungkin untuk jenis `Watch` acara dengan `SUM(Items.LENGTH)` fungsi.

Anda dapat menerapkan `SUM()` fungsi hanya ke kolom numerik kumpulan data interaksi Item dan Item. Untuk menerapkan `SUM()` fungsi ke kolom dalam kumpulan data Item, Anda harus terlebih dahulu mengimpor metadata item.

Anda dapat membuat atribusi metrik dengan konsol Amazon Personalize AWS Command Line Interface, AWS atau SDKS.

Topik

- Membuat atribusi metrik (konsol)
- Membuat atribusi metrik ()AWS CLI
- Membuat atribusi metrik ()AWS SDKs

Membuat atribusi metrik (konsol)

Untuk membuat atribusi metrik dengan konsol Amazon Personalize, Anda menavigasi ke halaman atribusi Metrik dan memilih Buat atribusi metrik. Saat membuat atribusi metrik, Anda menentukan jalur bucket Amazon S3 opsional, peran layanan Amazon Personalisasi IAM, dan daftar metrik yang akan dilaporkan.

Saat membuat pekerjaan impor kumpulan data interaksi Item dengan konsol Amazon Personalize, Anda memiliki opsi untuk membuat atribusi metrik di tab baru. Kemudian Anda dapat kembali ke pekerjaan impor untuk menyelesaikannya. Jika Anda sudah berada di halaman Konfigurasi atribusi metrik, Anda dapat melompat ke langkah 4.

Untuk membuat atribusi metrik

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.

2. Pilih grup kumpulan data Anda.
3. Di panel navigasi, di bawah Sumber daya khusus, pilih Atribusi metrik.
4. Di detail atribusi metrik, pilih Buat atribusi metrik.
5. Pada halaman Konfigurasi atribusi metrik, beri nama atribusi metrik.
6. Jika Anda ingin mempublikasikan metrik ke Amazon S3 untuk jalur keluaran data Amazon S3, masukkan bucket Amazon S3 tujuan. Ini memungkinkan opsi untuk mempublikasikan metrik setiap kali Anda membuat pekerjaan impor kumpulan data. Gunakan sintaks berikut:

   `s3://amzn-s3-demo-bucket/<folder> path>`
7. Jika Anda menggunakan AWS KMS untuk enkripsi, untuk ARN kunci KMS, masukkan Nama Sumber Daya Amazon (ARN) untuk kunci tersebut. AWS KMS Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.
8. Dalam peran IAM, pilih untuk membuat peran layanan baru atau menggunakan yang sudah ada. Peran yang Anda pilih harus memiliki `PutMetricData` izin untuk CloudWatch. Jika Anda ingin mempublikasikan ke Amazon S3, peran tersebut harus memiliki `PutObject` izin untuk bucket Amazon S3 Anda.

   Untuk menggunakan peran yang Anda buatMembuat peran IAM untuk Amazon Personalize, Anda mungkin harus menambahkan kebijakan untuk CloudWatch dan Amazon S3.

   Untuk contoh kebijakan, lihat Memberikan Amazon Personalisasi akses ke CloudWatch danMemberikan Amazon Personalisasi akses ke bucket Amazon S3 Anda.
9. Pilih Berikutnya.
10. Pada halaman Tentukan atribut metrik, pilih cara menentukan metrik. Pilih Buat atribut metrik untuk menggunakan alat pembuat. Pilih Atribut metrik masukan untuk memasukkan metrik dalam format JSON.

    - Jika Anda memilih Membangun atribut metrik, untuk setiap metrik berikan nama, jenis peristiwa, dan pilih fungsi. Untuk `SUM()` fungsi, pilih nama kolom. Pilih Tambahkan atribut metrik untuk menambahkan metrik tambahan.
    - Jika Anda memilih Atribut metrik masukan, masukkan setiap metrik dalam format JSON. Berikut ini menunjukkan cara memformat metrik.

      ```
      {
      ```

```
    "EventType": "watch",
    "MetricName": "MinutesWatchedTracker",
    "MetricMathExpression": "SUM(Items.LENGTH)"
}
```

11. Pilih Berikutnya.
12. Pada halaman Tinjau dan buat, tinjau detail atribusi metrik baru. Untuk melakukan perubahan, pilih Sebelumnya. Untuk membuat atribusi metrik, pilih Buat. Saat atribusi metrik aktif, Anda dapat mulai mengimpor data dan melihat hasilnya. Untuk informasi tentang melihat hasil, lihatMelihat grafik data metrik di CloudWatch. Untuk informasi tentang mempublikasikan hasil ke Amazon S3, lihat. Menerbitkan laporan atribusi metrik ke Amazon S3

Membuat atribusi metrik ()AWS CLI

Kode berikut menunjukkan cara membuat atribusi metrik dengan. AWS Command Line Interface Peran yang Anda tentukan harus memiliki `PutMetricData` izin untuk CloudWatch dan, jika memublikasikan ke Amazon S3`PutObject`, izin untuk bucket Amazon S3 Anda. Untuk menggunakan peran yang Anda buatMembuat peran IAM untuk Amazon Personalize, Anda mungkin harus menambahkan kebijakan untuk CloudWatch dan Amazon S3. Untuk contoh kebijakan, lihat Memberikan Amazon Personalisasi akses ke CloudWatch danMemberikan Amazon Personalisasi akses ke bucket Amazon S3 Anda.

Untuk setiap metrik tentukan nama, jenis peristiwa, dan ekspresi (fungsi). Fungsi yang tersedia termasuk `SUM(DatasetType.COLUMN_NAME)` dan`SAMPLECOUNT()`. Untuk fungsi SUM (), tentukan jenis kumpulan data dan nama kolom. Misalnya, `SUM(Items.LENGTH)`. Untuk informasi tentang setiap parameter, lihatCreateMetricAttribution.

```
aws personalize create-metric-attribution \
--name metric attribution name \
--dataset-group-arn dataset group arn \
--metrics-output-config "{\"roleArn\": \"Amazon Personalize service role ARN\",
 \"s3DataDestination\":{\"kmsKeyArn\":\"kms key ARN\",\"path\":\"s3://amzn-s3-demo-
bucket/folder-name/\"}}" \
--metrics "[{
  \"eventType\": \"event type\",
  \"expression\": \"SUM(DatasetType.COLUMN_NAME)\",
  \"metricName\": \"metric name\"
}]"
```

Membuat atribusi metrik ()AWS SDKs

Kode berikut menunjukkan cara membuat atribusi metrik dengan SDK for Python (Boto3). Peran yang Anda tentukan harus memiliki `PutMetricData` izin untuk CloudWatch dan, jika memublikasikan ke Amazon S3`PutObject`, izin untuk bucket Amazon S3 Anda. Untuk menggunakan peran yang Anda buatMembuat peran IAM untuk Amazon Personalize, Anda mungkin harus menambahkan kebijakan untuk CloudWatch dan Amazon S3. Untuk contoh kebijakan, lihat Memberikan Amazon Personalisasi akses ke CloudWatch danMemberikan Amazon Personalisasi akses ke bucket Amazon S3 Anda.

Untuk setiap metrik tentukan nama, jenis peristiwa, dan ekspresi (fungsi). Fungsi yang tersedia termasuk `SUM(DatasetType.COLUMN_NAME)` dan`SAMPLECOUNT()`. Untuk fungsi SUM (), tentukan jenis kumpulan data dan nama kolom. Misalnya, `SUM(Items.LENGTH)`. Untuk informasi tentang setiap parameter, lihatCreateMetricAttribution.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

metricsList = [{
      "eventType": "event type",
      "expression": "SUM(DatasetType.COLUMN_NAME)",
      "metricName": "metric name"
}]

outputConfig = {
  "roleArn": "Amazon Personalize service role ARN",
  "s3DataDestination": {
    "kmsKeyArn": "key ARN",
    "path": "s3://amzn-s3-demo-bucket/<folder>"
  }
}
response = personalize.create_metric_attribution(
  name = 'metric attribution name',
  datasetGroupArn = 'dataset group arn',
  metricsOutputConfig = outputConfig,
  metrics = metricsList
)

metric_attribution_arn = response['metricAttributionArn']
```

```
print ('Metric attribution ARN: ' + metric_attribution_arn)

description = personalize.describe_metric_attribution(
    metricAttributionArn = metric_attribution_arn)['metricAttribution']

print('Name: ' + description['name'])
print('ARN: ' + description['metricAttributionArn'])
print('Status: ' + description['status'])
```

SDK for Java 2.x

```
public static String createMetricAttribution(PersonalizeClient personalizeClient,
                                              String eventType,
                                              String expression,
                                              String metricName,
                                              String metricAttributionName,
                                              String roleArn,
                                              String s3Path,
                                              String kmsKeyArn,
                                              String datasetGroupArn) {
    String metricAttributionArn = "";

    try {

        MetricAttribute attribute = MetricAttribute.builder()
                .eventType(eventType)
                .expression(expression)
                .metricName(metricName)
                .build();

        ArrayList<MetricAttribute> metricAttributes = new ArrayList<>();
        metricAttributes.add(attribute);

        S3DataConfig s3DataDestination = S3DataConfig.builder()
                .kmsKeyArn(kmsKeyArn)
                .path(s3Path)
                .build();

        MetricAttributionOutput outputConfig = MetricAttributionOutput.builder()
                .roleArn(roleArn)
                .s3DataDestination(s3DataDestination)
                .build();
```

```
        CreateMetricAttributionRequest createMetricAttributionRequest =
 CreateMetricAttributionRequest.builder()
                .name(metricAttributionName)
                .datasetGroupArn(datasetGroupArn)
                .metrics(metricAttributes)
                .metricsOutputConfig(outputConfig)
                .build();
        CreateMetricAttributionResponse createMetricAttributionResponse =
 personalizeClient.createMetricAttribution(createMetricAttributionRequest);

        metricAttributionArn =
 createMetricAttributionResponse.metricAttributionArn();
        System.out.println("Metric attribution ARN: " + metricAttributionArn);
        return metricAttributionArn;
    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { CreateMetricAttributionCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

// set the metric attribution param
export const createMetricAttributionParam = {
  name: "METRIC_ATTRIBUTION_NAME",            /* required */
  datasetGroupArn: "DATASET_GROUP_ARN",       /* required */
  metricsOutputConfig: {
    roleArn: "ROLE_ARN",                      /* required */
    s3DataDestination: {
      kmsKeyArn: "KEY_ARN",                                                  /*
 optional */
      path: "s3://amzn-s3-demo-bucket/<folderName>/",    /* optional */
    },
  },
```

```
  metrics: [
    {
      eventType: "EVENT_TYPE",                      /* required for each metric */
      expression: "SUM(DatasetType.COLUMN_NAME)",   /* required for each metric */
      metricName: "METRIC_NAME",                    /* required for each metric */
    }
  ]
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateMetricAttributionCommand(createMetricAttributionParam)
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Memperbarui atribusi metrik Amazon Personalisasi

Saat memperbarui atribusi metrik, Anda dapat menambahkan dan menghapus metrik dan memodifikasi konfigurasi outputnya. Anda dapat memperbarui atribusi metrik dengan konsol Amazon Personalize AWS Command Line Interface, AWS atau SDKS.

Topik

- Memperbarui atribusi metrik (konsol)
- Memperbarui atribusi metrik ()AWS CLI
- Memperbarui atribusi metrik (AWS SDK)

Memperbarui atribusi metrik (konsol)

Untuk memperbarui atribusi metrik dengan konsol Amazon Personalize, Anda membuat perubahan di halaman atribusi Metrik.

Untuk memperbarui atribusi metrik

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data Anda.
3. Di panel navigasi, pilih Atribusi metrik.
4. Di bagian bawah, pilih tab Atribut metrik atau tab Konfigurasi atribusi metrik untuk mulai membuat perubahan.

   - Untuk menambah atau menghapus metrik, pilih tab Atribut metrik dan pilih Edit atribut. Buat perubahan di halaman Edit atribut metrik dan pilih Perbarui untuk menyimpan perubahan Anda.
   - Untuk membuat perubahan pada bucket keluaran Amazon S3 atau peran layanan IAM, pilih tab Edit konfigurasi atribusi metrik dan buat perubahan pada halaman konfigurasi Edit atribusi. Pilih Perbarui untuk menyimpan perubahan.

Memperbarui atribusi metrik ()AWS CLI

Setelah membuat atribusi metrik, Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk menambah dan menghapus metrik dan memodifikasi konfigurasi outputnya. Kode berikut menunjukkan cara menghapus metrik dengan `update-metric-attribution` perintah:

```
aws personalize update-metric-attribution \
--metric-attribution-arn metric attribution arn \
--remove-metrics metricName1 metricName2
```

Kode berikut menunjukkan cara menambahkan metrik tambahan dan menentukan konfigurasi keluaran baru:

```
aws personalize update-metric-attribution \
--metric-attribution-arn metric attribution arn \
--metrics-output-config "{\"roleArn\": \"new role ARN\", \"s3DataDestination\":
{\"kmsKeyArn\":\"kms key ARN\",\"path\":\"s3://amzn-s3-demo-bucket2/new-folder-name/
\"}}" \
--add-metrics "[{
  \"eventType\": \"event type\",
  \"expression\": \"SUM(DatasetType.COLUMN_NAME)\",
  \"metricName\": \"metric name\"
```

```
}]"
```

Jika berhasil, Amazon Personalize mengembalikan ARN atribusi metrik yang Anda perbarui. Untuk daftar lengkap semua parameter, lihatUpdateMetricAttribution.

Memperbarui atribusi metrik (AWS SDK)

Setelah membuat atribusi metrik, Anda dapat menambahkan atau menghapus metrik dan mengubah konfigurasi outputnya. Kode berikut menunjukkan cara menghapus metrik dari atribusi metrik.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

metricsToRemove = ["metricName1", "metricName2"]

response = personalize.update_metric_attribution(
  metricAttributionArn = "metric attribution ARN",
  removeMetrics = metricsToRemove
)
```

SDK for Java 2.x

```
public static void removeMetrics(PersonalizeClient client,
                                 String metricAttributionArn,
                                 String metric1Name,
                                 String metric2Name) {

    ArrayList<String> metricsToRemove = new ArrayList<>(Arrays.asList(metric1Name,
 metric2Name));

    try {

        UpdateMetricAttributionRequest request =
 UpdateMetricAttributionRequest.builder()
                .metricAttributionArn(metricAttributionArn)
                .removeMetrics(metricsToRemove)
                .build();

        UpdateMetricAttributionResponse response =
 client.updateMetricAttribution(request);
```

```
        System.out.println(response);

    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
}
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import {UpdateMetricAttributionCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

// set the update request param
export const updateMetricAttributionParam = {
  metricAttributionArn: "METRIC_ATTRIBUTION_ARN",    /* required */
  removeMetrics: ["METRIC_NAME_1", "METRIC_NAME_2"]    /* specify list of names of
 metrics to delete */
};
export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new UpdateMetricAttributionCommand(updateMetricAttributionParam)
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Kode berikut menunjukkan cara menambahkan metrik tambahan dan menentukan konfigurasi keluaran baru:

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

newMetrics = [{
      "eventType": "event type",
      "expression": "SUM(DatasetType.COLUMN_NAME)",
      "metricName": "metric name"
}]

newOutputConfig = {
  "roleArn": "Amazon Personalize service role ARN",
  "s3DataDestination": {
    "kmsKeyArn": "key ARN",
    "path": "s3://amzn-s3-demo-bucket/<folder>"
  }
}

response = personalize.update_metric_attribution(
  metricAttributionArn = "metric attribution arn",
  metricsOutputConfig = newOutputConfig,
  addMetrics = newMetrics
)
```

SDK for Java 2.x

```
public static void addMetricsAndUpdateOutputConfig(PersonalizeClient
 personalizeClient,
                                                    String metricAttributionArn,
                                                    String newMetric1EventType,
                                                    String newMetric1Expression,
                                                    String newMetric1Name,
                                                    String newMetric2EventType,
                                                    String newMetric2Expression,
                                                    String newMetric2Name,
                                                    String roleArn,
                                                    String s3Path,
                                                    String kmsKeyArn) {
    try {

        MetricAttribute newAttribute = MetricAttribute.builder()
```

```
                .eventType(newMetric1EventType)
                .expression(newMetric1Expression)
                .metricName(newMetric1Name)
                .build();

        MetricAttribute newAttribute2 = MetricAttribute.builder()
                .eventType(newMetric2EventType)
                .expression(newMetric2Expression)
                .metricName(newMetric2Name)
                .build();

        ArrayList<MetricAttribute> newAttributes = new
 ArrayList<>(Arrays.asList(newAttribute, newAttribute2));

        S3DataConfig newDataDestination = S3DataConfig.builder()
                .kmsKeyArn(kmsKeyArn)
                .path(s3Path)
                .build();

        MetricAttributionOutput newOutputConfig = MetricAttributionOutput.builder()
                .roleArn(roleArn)
                .s3DataDestination(newDataDestination)
                .build();

        UpdateMetricAttributionRequest request =
 UpdateMetricAttributionRequest.builder()
                .metricAttributionArn(metricAttributionArn)
                .metricsOutputConfig(newOutputConfig)
                .addMetrics(newAttributes)
                .build();

        UpdateMetricAttributionResponse response =
 personalizeClient.updateMetricAttribution(request);
        System.out.println("New metrics added!");
        System.out.println(response);

    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
}
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import {UpdateMetricAttributionCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

export const updateMetricAttributionParam = {
  metricAttributionArn: "METRIC_ATTRIBUTION_ARN",
  addMetrics: [
    {
      eventType: "EVENT_TYPE",                      /* required for each metric */
      expression: "SUM(DatasetType.COLUMN_NAME)",   /* required for each metric */
      metricName: "METRIC_NAME",                    /* required for each metric */
    }
  ],
  metricsOutputConfig: {
    roleArn: "ROLE_ARN",                     /* required */
    s3DataDestination: {
      kmsKeyArn: "KEY_ARN",                                                  /*
 optional */
      path: "s3://amzn-s3-demo-bucket/<folderName>/",    /* optional */
    },
  }
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new UpdateMetricAttributionCommand(updateMetricAttributionParam)
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Jika berhasil, Amazon Personalize mengembalikan ARN atribusi metrik yang Anda perbarui. Untuk daftar lengkap semua parameter, lihatUpdateMetricAttribution.

Menghapus atribusi metrik Amazon Personalisasi

Jika Anda tidak lagi ingin membuat laporan, Anda dapat menghapus atribusi metrik. Menghapus atribusi metrik akan menghapus semua metrik dan konfigurasi outputnya.

Jika Anda menghapus atribusi metrik, Amazon Personalize akan berhenti mengirim laporan terkait dan data PutEvents massal tambahan secara otomatis. CloudWatch Data yang sudah dikirim CloudWatch atau dipublikasikan ke Amazon S3 tidak terpengaruh. Anda dapat menghapus atribusi metrik dengan konsol Amazon Personalize AWS Command Line Interface, AWS atau SDKS.

Topik

- Menghapus atribusi metrik (konsol)
- Menghapus atribusi metrik ()AWS CLI
- Menghapus atribusi metrik ()AWS SDKs

Menghapus atribusi metrik (konsol)

Anda menghapus atribusi metrik pada halaman ikhtisar untuk atribusi metrik Anda.

Untuk menghapus atribusi metrik

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih grup kumpulan data Anda.
3. Di panel navigasi, pilih Atribusi metrik.
4. Pilih Hapus dan kemudian konfirmasikan penghapusan.

Menghapus atribusi metrik ()AWS CLI

Untuk menghapus atribusi metrik dengan AWS CLI, gunakan `delete-metric-attribution` perintah sebagai berikut.

```
aws personalize delete-metric-attribution --metric-attribution-arn metric attribution
 ARN
```

Menghapus atribusi metrik ()AWS SDKs

Kode berikut menunjukkan cara menghapus atribusi metrik dengan SDK for Python (Boto3):

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.delete_metric_attribution(
  metricAttributionArn = 'metric attribution ARN'
)
```

SDK for Java 2.x

```
public static void deleteMetricAttribution(PersonalizeClient client, String
 metricAttributionArn) {

    try {

        DeleteMetricAttributionRequest request =
 DeleteMetricAttributionRequest.builder()
                .metricAttributionArn(metricAttributionArn)
                .build();

        DeleteMetricAttributionResponse response =
 client.deleteMetricAttribution(request);
        if (response.sdkHttpResponse().statusCode() == 200) {
            System.out.println("Metric attribution deleted!");
        }

    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
}
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { DeleteMetricAttributionCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";
```

```
// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

export const deleteMetricAttributionParam = {
  metricAttributionArn: "METRIC_ATTRIBUTION_ARN",
};
export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new DeleteMetricAttributionCommand(deleteMetricAttributionParam)
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Melihat grafik data metrik di CloudWatch

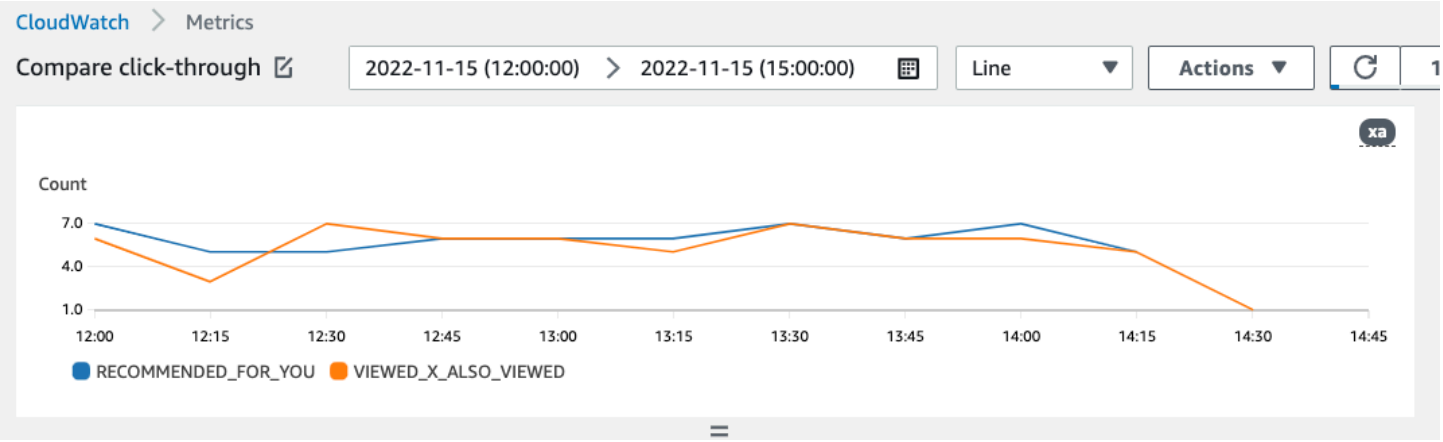
> **Important**
>
> Setelah Anda membuat atribusi metrik dan merekam peristiwa atau mengimpor data massal tambahan, Anda akan dikenakan CloudWatch biaya bulanan per metrik. Untuk informasi tentang CloudWatch harga, lihat halaman CloudWatch harga Amazon. Untuk berhenti mengirim metrik ke CloudWatch, hapus atribusi metrik.

Setelah Anda membuat atribusi metrik, Amazon Personalize secara otomatis mengirimkan metrik PutEvents dari dan data massal tambahan ke Amazon. CloudWatch Anda dapat memilih metrik dan membuat grafik data metrik menggunakan konsol. CloudWatch Grafik ini dapat membantu Anda secara visual memeriksa dan membandingkan kinerja dan dampak dari berbagai pemberi rekomendasi atau kampanye.

Untuk membandingkan sumber, setiap peristiwa interaksi harus menyertakan `recommendationId` atau`eventAttributionSource`. Untuk contoh kode yang menunjukkan cara menyertakan data ini dalam suatu peristiwa, lihatMetrik peristiwa dan laporan atribusi.

Untuk melihat metrik CloudWatch, selesaikan prosedur yang ditemukan di Grafik metrik. Anda dapat melihat data pada berbagai tingkat detail. Periode minimum yang dapat Anda buat grafik adalah 15 menit. Anda dapat melihat data Amazon Personalisasi dari 2 minggu sebelumnya CloudWatch — data lama diabaikan. Untuk istilah penelusuran, tentukan nama yang Anda berikan metrik saat Anda membuat atribusi metrik.

Berikut ini adalah contoh bagaimana metrik mungkin muncul CloudWatch. Metrik menunjukkan rasio klik-tayang untuk setiap 15 menit untuk dua pemberi rekomendasi yang berbeda.



Menerbitkan laporan atribusi metrik ke Amazon S3

Untuk semua data massal, jika Anda menyediakan bucket Amazon S3 saat membuat atribusi metrik, Anda dapat memilih untuk menerbitkan laporan metrik ke bucket Amazon S3 setiap kali Anda membuat pekerjaan impor kumpulan data untuk data interaksi.

Untuk memublikasikan metrik ke Amazon S3, Anda menyediakan jalur ke bucket Amazon S3 di atribusi metrik. Kemudian Anda memublikasikan laporan ke Amazon S3 saat Anda membuat pekerjaan impor kumpulan data. Saat pekerjaan selesai, Anda dapat menemukan metrik di bucket Amazon S3 Anda. Setiap kali Anda mempublikasikan metrik, Amazon Personalize membuat file baru di bucket Amazon S3 Anda. Nama file termasuk metode impor dan tanggal sebagai berikut:

`AggregatedAttributionMetrics - ImportMethod - Timestamp.csv`

Berikut ini adalah contoh bagaimana beberapa baris pertama dari file CSV laporan metrik mungkin muncul. Metrik dalam contoh ini melaporkan total klik dari dua pemberi rekomendasi berbeda selama interval 15 menit. Setiap pemberi rekomendasi diidentifikasi dengan Nama Sumber Daya Amazon (ARN) di kolom EVENT_ATTRIBUTION_SOURCE.

```
METRIC_NAME,EVENT_TYPE,VALUE,MATH_FUNCTION,EVENT_ATTRIBUTION_SOURCE,TIMESTAMP
```

```
COUNTWATCHES,WATCH,12.0,samplecount,arn:aws:personalize:us-west-2:acctNum:recommender/
recommender1Name,1666925124
COUNTWATCHES,WATCH,112.0,samplecount,arn:aws:personalize:us-west-2:acctNum:recommender/
recommender2Name,1666924224
COUNTWATCHES,WATCH,10.0,samplecount,arn:aws:personalize:us-west-2:acctNum:recommender/
recommender1Name,1666924224
COUNTWATCHES,WATCH,254.0,samplecount,arn:aws:personalize:us-west-2:acctNum:recommender/
recommender2Name,1666922424
COUNTWATCHES,WATCH,112.0,samplecount,arn:aws:personalize:us-west-2:acctNum:recommender/
recommender1Name,1666922424
COUNTWATCHES,WATCH,100.0,samplecount,arn:aws:personalize:us-west-2:acctNum:recommender/
recommender2Name,1666922424
......
.....
```

Menerbitkan metrik untuk data massal ke Amazon S3 (konsol)

Untuk memublikasikan metrik ke bucket Amazon S3 dengan konsol Amazon Personalize, buat tugas impor kumpulan data dan pilih Publikasikan metrik untuk pekerjaan impor ini di Publikasikan metrik peristiwa ke S3.

Untuk step-by-step instruksi, lihatMembuat pekerjaan impor dataset (konsol).

Menerbitkan metrik untuk data massal ke Amazon S3 ()AWS CLI

Untuk memublikasikan metrik ke bucket Amazon S3 dengan AWS Command Line Interface AWS CLI(), gunakan kode berikut untuk membuat tugas impor kumpulan data dan memberikan tanda. `publishAttributionMetricsToS3` Jika Anda tidak ingin mempublikasikan metrik untuk pekerjaan tertentu, hilangkan bendera. Untuk informasi tentang setiap parameter, lihatCreateDatasetImportJob.

```
aws personalize create-dataset-import-job \
--job-name dataset import job name \
--dataset-arn dataset arn \
--data-source dataLocation=s3://amzn-s3-demo-bucket/filename \
--role-arn roleArn \
--import-mode INCREMENTAL \
--publish-attribution-metrics-to-s3
```

Menerbitkan metrik untuk data massal ke Amazon S3 ()AWS SDKs

Untuk memublikasikan metrik ke bucket Amazon S3 dengan AWS SDKs, buat tugas impor kumpulan data dan `publishAttributionMetricsToS3` setel ke true. Untuk informasi tentang setiap parameter, lihatCreateDatasetImportJob.

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.create_dataset_import_job(
    jobName = 'YourImportJob',
    datasetArn = 'dataset_arn',
    dataSource = {'dataLocation':'s3://amzn-s3-demo-bucket/file.csv'},
    roleArn = 'role_arn',
    importMode = 'INCREMENTAL',
    publishAttributionMetricsToS3 = True
)

dsij_arn = response['datasetImportJobArn']

print ('Dataset Import Job arn: ' + dsij_arn)

description = personalize.describe_dataset_import_job(
    datasetImportJobArn = dsij_arn)['datasetImportJob']

print('Name: ' + description['jobName'])
print('ARN: ' + description['datasetImportJobArn'])
print('Status: ' + description['status'])
```

SDK for Java 2.x

```
public static String createPersonalizeDatasetImportJob(PersonalizeClient
 personalizeClient,
                                                        String jobName,
                                                        String datasetArn,
                                                        String s3BucketPath,
                                                        String roleArn,
                                                        ImportMode importMode,
                                                        boolean publishToS3) {
```

```
 long waitInMilliseconds = 60 * 1000;
 String status;
 String datasetImportJobArn;

 try {
     DataSource importDataSource = DataSource.builder()
             .dataLocation(s3BucketPath)
             .build();

     CreateDatasetImportJobRequest createDatasetImportJobRequest =
CreateDatasetImportJobRequest.builder()
             .datasetArn(datasetArn)
             .dataSource(importDataSource)
             .jobName(jobName)
             .roleArn(roleArn)
             .importMode(importMode)
             .publishAttributionMetricsToS3(publishToS3)
             .build();

     datasetImportJobArn =
personalizeClient.createDatasetImportJob(createDatasetImportJobRequest)
             .datasetImportJobArn();

     DescribeDatasetImportJobRequest describeDatasetImportJobRequest =
DescribeDatasetImportJobRequest.builder()
             .datasetImportJobArn(datasetImportJobArn)
             .build();

     long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

     while (Instant.now().getEpochSecond() < maxTime) {

         DatasetImportJob datasetImportJob = personalizeClient
                 .describeDatasetImportJob(describeDatasetImportJobRequest)
                 .datasetImportJob();

         status = datasetImportJob.status();
         System.out.println("Dataset import job status: " + status);

         if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
             break;
         }
         try {
             Thread.sleep(waitInMilliseconds);
```

```
            } catch (InterruptedException e) {
                System.out.println(e.getMessage());
            }
        }
        return datasetImportJobArn;

    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return "";
}
```

SDK for JavaScript v3

```
// Get service clients and commands using ES6 syntax.
import { CreateDatasetImportJobCommand, PersonalizeClient } from
  "@aws-sdk/client-personalize";

// create personalizeClient
const personalizeClient = new PersonalizeClient({
  region: "REGION"
});

// Set the dataset import job parameters.
export const datasetImportJobParam = {
  datasetArn: 'DATASET_ARN', /* required */
  dataSource: {
    dataLocation: 's3://amzn-s3-demo-bucket/<folderName>/<CSVfilename>.csv'  /*
 required */
  },
  jobName: 'NAME',                        /* required */
  roleArn: 'ROLE_ARN',                    /* required */
  importMode: "FULL",                     /* optional, default is FULL */
  publishAttributionMetricsToS3: true     /* set to true to publish metrics to
 Amazon S3 bucket */
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(new
 CreateDatasetImportJobCommand(datasetImportJobParam));
    console.log("Success", response);
    return response; // For unit tests.
```

```
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Mengukur dampak rekomendasi dengan A/B pengujian

Melakukan A/B tes terdiri dari menjalankan eksperimen dengan beberapa variasi dan membandingkan hasilnya. Melakukan A/B pengujian dengan rekomendasi Amazon Personalize melibatkan menampilkan berbagai kelompok pengguna berbagai jenis rekomendasi dan kemudian membandingkan hasilnya. Anda dapat menggunakan A/B pengujian untuk membantu membandingkan dan mengevaluasi berbagai strategi rekomendasi, dan mengukur dampak rekomendasi.

Misalnya, Anda dapat menggunakan A/B pengujian untuk melihat apakah rekomendasi Amazon Personalisasi meningkatkan rasio klik-tayang. Untuk menguji skenario ini, Anda dapat menampilkan satu grup rekomendasi pengguna yang tidak dipersonalisasi, seperti produk unggulan. Dan Anda mungkin menunjukkan rekomendasi yang dipersonalisasi grup lain yang dihasilkan oleh Amazon Personalize. Saat pelanggan berinteraksi dengan item, Anda dapat mencatat hasilnya dan melihat strategi mana yang menghasilkan rasio klik-tayang tertinggi.

Alur kerja untuk melakukan A/B pengujian dengan rekomendasi Amazon Personalize adalah sebagai berikut:

1. Rencanakan eksperimen Anda — Tentukan hipotesis yang dapat diukur, identifikasi tujuan bisnis, tentukan variasi eksperimen, dan tentukan kerangka waktu eksperimen Anda.
2. Pisahkan pengguna Anda — Pisahkan pengguna menjadi dua grup atau lebih, dengan grup kontrol dan satu atau beberapa grup eksperimen.
3. Jalankan eksperimen Anda — Tampilkan rekomendasi modifikasi grup eksperimen kepada pengguna. Tampilkan pengguna dalam rekomendasi grup kontrol tanpa perubahan. Rekam interaksi mereka dengan rekomendasi untuk melacak hasil.
4. Mengevaluasi hasil — Menganalisis hasil eksperimen untuk menentukan apakah modifikasi membuat perbedaan yang signifikan secara statistik untuk kelompok eksperimen.

Anda dapat menggunakan Amazon CloudWatch Terbukti untuk melakukan A/B pengujian dengan rekomendasi Amazon Personalize. Dengan CloudWatch Evidently, Anda dapat menentukan

eksperimen, melacak indikator kinerja utama (KPIs), merutekan lalu lintas permintaan rekomendasi ke sumber daya Amazon Personalize yang relevan, dan mengevaluasi hasil eksperimen. Untuk informasi selengkapnya, lihat Pengujian A/B dengan CloudWatch Terbukti.

Topik

- A/B menguji praktik terbaik
- Pengujian A/B dengan CloudWatch Terbukti

A/B menguji praktik terbaik

Gunakan praktik terbaik berikut untuk membantu Anda merancang dan memelihara A/B pengujian untuk rekomendasi Amazon Personalisasi.

- Identifikasi tujuan bisnis yang dapat diukur. Verifikasi bahwa rekomendasi berbeda yang ingin Anda bandingkan keduanya selaras dengan tujuan bisnis ini dan tidak terkait dengan tujuan yang berbeda atau tidak dapat diukur.
- Tentukan hipotesis terukur yang sejalan dengan tujuan bisnis Anda. Misalnya, Anda dapat memprediksi bahwa promosi untuk konten yang dibuat khusus Anda sendiri akan menghasilkan 20% lebih banyak klik dari item ini. Hipotesis Anda menentukan modifikasi yang Anda buat untuk kelompok eksperimen Anda.
- Tentukan indikator kinerja kunci yang relevan (KPIs) yang terkait dengan hipotesis Anda. Anda gunakan KPIs untuk mengukur hasil eksperimen Anda. Ini mungkin sebagai berikut:
  - Rasio klik-tayang
  - Waktu menonton
  - Harga total
- Verifikasi bahwa jumlah total pengguna dalam percobaan cukup besar untuk mencapai hasil yang signifikan secara statistik, tergantung pada hipotesis Anda.
- Tentukan strategi pemisahan lalu lintas Anda sebelum memulai eksperimen. Hindari mengubah pemisahan lalu lintas saat percobaan sedang berjalan.
- Jaga agar pengalaman pengguna aplikasi atau situs web Anda tetap sama untuk grup eksperimen dan grup kontrol Anda, kecuali untuk modifikasi yang terkait dengan eksperimen Anda (misalnya, model). Variasi dalam pengalaman pengguna, seperti UI atau latensi, dapat menyebabkan hasil yang menyesatkan.
- Kontrol faktor eksternal, seperti hari libur, kampanye pemasaran yang sedang berlangsung, dan batasan browser. Faktor-faktor eksternal ini dapat menyebabkan hasil yang menyesatkan.

- Hindari mengubah rekomendasi Amazon Personalize kecuali terkait langsung dengan hipotesis atau persyaratan bisnis Anda. Perubahan seperti menerapkan filter atau mengubah urutan secara manual dapat menyebabkan hasil yang menyesatkan.
- Ketika Anda mengevaluasi hasil, pastikan bahwa hasilnya signifikan secara statistik sebelum menarik kesimpulan. Standar industri adalah tingkat signifikansi 5%.

Pengujian A/B dengan CloudWatch Terbukti

Setelah membuat pemberi rekomendasi atau menerapkan versi solusi khusus dengan kampanye, Anda dapat melakukan A/B pengujian dengan rekomendasi Amazon Personalisasi dan Amazon Jelas. CloudWatch Video berikut menjelaskan proses penggunaan CloudWatch Evidently untuk melakukan pengujian A/B dengan rekomendasi Amazon Personalize. Untuk step-by-step instruksi, lihatMelakukan A/B tes dengan CloudWatch Evidently.

Lakukan Pengujian AB dengan Amazon Personalize dan Jelas CloudWatch

Topik

- Melakukan A/B tes dengan CloudWatch Evidently
- Contoh implementasi

Melakukan A/B tes dengan CloudWatch Evidently

Untuk melakukan A/B pengujian dengan Amazon Personalize dan Amazon CloudWatch Evidently, buat proyek CloudWatch Evidently, tentukan fitur dan variasinya, perbarui aplikasi Anda untuk mendukung eksperimen Anda, dan buat serta jalankan eksperimen. Saat eksperimen berjalan, Anda dapat melihat hasilnya di CloudWatch Evidently.

Untuk melakukan A/B tes dengan Amazon Personalize and Evidently CloudWatch

1. Buat proyek yang CloudWatch jelas. Proyek adalah pengelompokan CloudWatch sumber daya yang logis. Dalam proyek tersebut, Anda membuat fitur yang memiliki variasi yang ingin Anda uji atau luncurkan. Untuk step-by-step petunjuknya, lihat Membuat proyek baru di Panduan CloudWatch Pengguna Amazon.
2. Tambahkan fitur ke proyek Anda dan tentukan variasinya. Untuk eksperimen ini, fitur Anda harus mewakili skenario rekomendasi yang ingin Anda uji, seperti rasio klik-tayang.

Saat menambahkan fitur, tentukan pengenal untuk memetakan variasi skenario yang berbeda ke Amazon Personalisasi pemberi rekomendasi atau kampanye khusus. Untuk setiap variasi, tentukan jenis Variasi, seperti String, beri nama variasi, dan berikan nilai.

Saat eksperimen Anda berjalan, aplikasi Anda menggunakan nilai variasi untuk menentukan sumber daya Amazon Personalize yang akan digunakan untuk rekomendasi. Misalnya, jika Anda menguji dua rekomendasi VIDEO_ON_DEMAND, satu dibuat untuk pilihan Teratas untuk kasus penggunaan Anda dan satu dibuat untuk kasus penggunaan Trending now, Anda dapat menetapkan JSON berikut sebagai Nilai untuk setiap variasi.

```
{"type":"top-picks-recommendations","arn":"arn:aws:personalize:us-west-2:<acct-
id>:recommender/top-picks-recommender"}
```

```
{"type":"trending-recommendations","arn":"arn:aws:personalize:us-west-2:<acct-
id>:recommender/trending-now-recommender"}
```

Anda dapat menentukan pengenal apa pun, selama aplikasi Anda dapat menggunakannya untuk mengidentifikasi sumber daya yang relevan. Misalnya, Anda mungkin hanya menentukan nama pemberi rekomendasi atau kampanye, dan membuat Amazon Resource Name (ARN) sumber daya dalam aplikasi Anda.

Untuk step-by-step petunjuk menambahkan fitur, lihat Menambahkan fitur ke proyek di Panduan CloudWatch Pengguna Amazon.

3. Perbarui aplikasi Anda untuk mendukung eksperimen Anda:

   - Evaluasi fitur - Gunakan operasi CloudWatch Evidently `EvaluateFeature` API untuk menetapkan variasi untuk setiap sesi pengguna. `EvaluateFeature`Respons mencakup nilai variasi yang Anda tentukan pada langkah sebelumnya. Dalam hal ini, ini adalah objek JSON dengan jenis pemberi rekomendasi dan itu adalah ARN dari pemberi rekomendasi. Perbarui kode permintaan rekomendasi Anda untuk mendapatkan rekomendasi dari sumber ini.

     Untuk informasi tentang mengevaluasi fitur, lihat Menggunakan EvaluateFeature di Panduan CloudWatch Pengguna Amazon.

   - Rekam hasil — Tambahkan kode ke aplikasi Anda untuk melacak hasil dari interaksi pengguna dengan rekomendasi.

Untuk melacak metrik eksperimen Anda di CloudWatch Evidently, gunakan operasi CloudWatch Evidently `PutProjectEvents` API untuk merekam hasil bagi setiap pengguna. Misalnya, jika pengguna dalam eksperimen mengklik item yang direkomendasikan, Anda akan mengirim detail untuk acara ini ke CloudWatch Evidently.

Untuk informasi tentang mengirim acara ke CloudWatch Evidently, lihat Menggunakan PutProjectEvents di Panduan CloudWatch Pengguna Amazon.

Untuk meningkatkan relevansi rekomendasi Amazon Personalize, Anda dapat merekam peristiwa hasil dengan operasi Amazon Personalize API. `PutEvents` Jika kasus penggunaan domain atau resep kustom Anda mendukung pembaruan real-time untuk rekomendasi, Amazon Personalize dapat belajar dari aktivitas terbaru pengguna Anda dan memperbarui rekomendasi saat mereka menggunakan aplikasi Anda. Jika tidak mendukung pembaruan, Amazon Personalize menggunakan data ini selama pelatihan ulang penuh berikutnya dari model Anda dan kemudian memengaruhi rekomendasi.

Untuk informasi tentang streaming acara ke Amazon Personalize, lihat. Merekam peristiwa waktu nyata untuk memengaruhi rekomendasi

4. Buat dan mulai percobaan. Saat Anda membuat eksperimen, tentukan yang berikut ini:

   - Fitur - Pilih fitur yang akan diuji dalam percobaan.
   - Audiens — Konfigurasikan berapa banyak pengguna Anda yang akan berpartisipasi, dan konfigurasikan cara membagi lalu lintas di antara variasi fitur.
   - Metrik — Tentukan metrik yang menentukan keberhasilan eksperimen. Misalnya, jumlah klik.

   Setelah Anda selesai membuat eksperimen, tentukan durasinya dan mulai percobaan. Untuk step-by-step petunjuk cara membuat dan memulai eksperimen di CloudWatch Evidently, lihat Membuat eksperimen di Panduan CloudWatch Pengguna Amazon.

5. Saat menjalankan eksperimen, Anda dapat melihat hasilnya di dasbor CloudWatch Evidently experiment. Untuk informasi tentang melihat hasil eksperimen, lihat Melihat hasil eksperimen di dasbor di Panduan CloudWatch Pengguna Amazon.

Contoh implementasi

Contoh implementasi berikut menunjukkan bagaimana menerapkan A/B pengujian dengan CloudWatch Evidently.

- Untuk contoh lengkap real-time APIs yang menyertakan kode sumber untuk mengimplementasikan A/B pengujian, lihat Personalisasi Waktu Nyata APIs di GitHub repositori AWS sampel.
- Untuk tutorial yang menjelaskan cara menggunakan A/B pengujian dengan CloudWatch Evidently dan contoh aplikasi react, lihat Tutorial: A/B testing with the Evidently sample application in the Amazon CloudWatch User Guide.

Personalisasi hasil pencarian dari OpenSearch

Anda dapat menggunakan Amazon Personalize untuk mempersonalisasi hasil dari open source atau OpenSearch Amazon OpenSearch Service untuk pengguna Anda.

OpenSearchadalah layanan pencarian sumber terbuka yang dikelola sendiri berdasarkan Lisensi Apache 2.0. Amazon OpenSearch Service adalah layanan terkelola yang membantu Anda menyebarkan, mengoperasikan, dan menskalakan OpenSearch sumber daya di AWS Cloud. Saat Anda menggunakan OpenSearch Layanan Amazon, OpenSearch mengambil dan memberi peringkat hasil.

Saat memberi peringkat hasil kueri, OpenSearch gunakan kerangka peringkat probabilistik yang disebut BM-25 untuk menghitung skor relevansi. Jika kata kunci yang berbeda muncul lebih sering dalam dokumen, BM-25 memberikan skor relevansi yang lebih tinggi untuk dokumen itu. OpenSearch peringkat tidak memperhitungkan perilaku pengguna seperti data klik-tayang.

Saat Anda menggunakan Amazon Personalize with OpenSearch, Amazon Personalize OpenSearch memberi peringkat ulang hasil berdasarkan perilaku pengguna sebelumnya, metadata apa pun tentang item, dan metadata apa pun tentang pengguna. OpenSearch kemudian memasukkan peringkat ulang sebelum mengembalikan respons pencarian ke aplikasi Anda. Anda mengontrol berapa banyak bobot yang OpenSearch memberikan peringkat dari Amazon Personalize saat menerapkannya ke OpenSearch hasil.

Dengan peringkat ulang ini, hasil dapat lebih menarik dan relevan dengan minat pengguna. Hal ini dapat menyebabkan peningkatan rasio klik-tayang dan rasio konversi untuk aplikasi Anda. Untuk contoh kasus penggunaan yang menjelaskan bagaimana pencarian yang dipersonalisasi dapat meningkatkan hasil untuk aplikasi e-commerce, lihatContoh kasus penggunaan.

Sebelum Anda mulai mempersonalisasi OpenSearch hasil, tinjau persyaratan yang tercantum diAmazon Personalisasi Persyaratan plugin Peringkat Pencarian.

Topik

- Contoh kasus penggunaan
- Cara kerja plugin Amazon Personalisasi Penelusuran Ranking
- Informasi tambahan
- Amazon Personalisasi Persyaratan plugin Peringkat Pencarian
- Personalisasi hasil dari OpenSearch Layanan Amazon dengan Amazon Personalize

- Mempersonalisasi hasil dari Open Search open source dengan Amazon Personalize
- Bidang untuk prosesor personalized_search_ranking respons
- Contoh metrik pipa

Contoh kasus penggunaan

Saat Anda menggunakan Amazon Personalize untuk menentukan peringkat ulang OpenSearch hasil, hasil penelusuran bisa lebih relevan bagi pengguna Anda. Misalnya, Anda mungkin memiliki aplikasi e-commerce yang menjual mobil. Jika pengguna Anda memasukkan kueri untuk mobil Toyota dan Anda tidak mempersonalisasi hasil, OpenSearch akan mengembalikan daftar mobil yang dibuat oleh Toyota berdasarkan kata kunci dalam data Anda. Daftar ini akan diberi peringkat dalam urutan yang sama untuk semua pengguna.

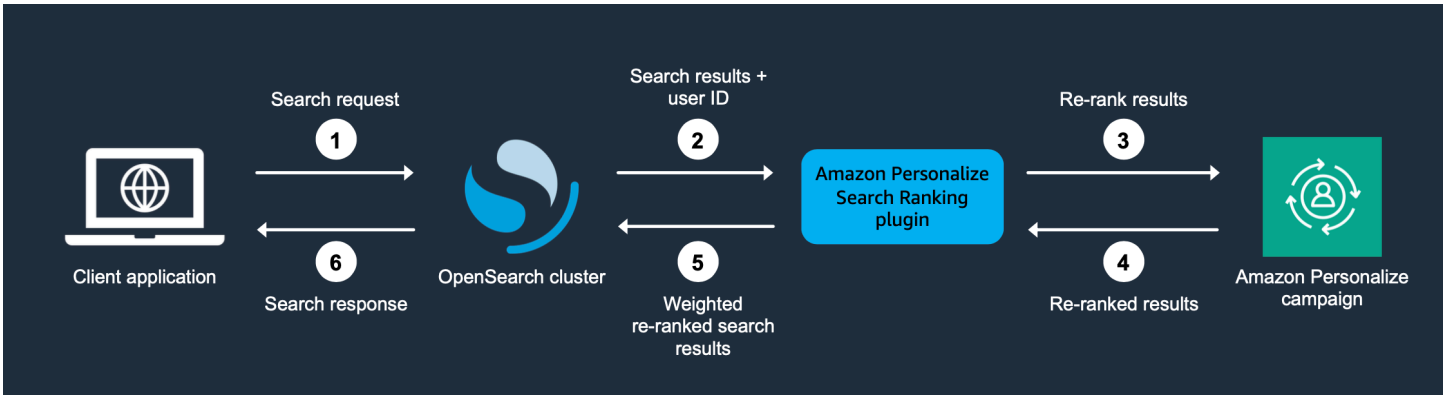
Tetapi jika Anda menggunakan Amazon Personalize untuk mempersonalisasi hasil, OpenSearch rangking ulang mobil-mobil ini dalam urutan relevansi bagi pengguna tertentu berdasarkan perilaku mereka—misalnya, klik mereka. Mobil yang paling mungkin diklik pengguna berada di peringkat pertama.

Ketika Anda mempersonalisasi OpenSearch hasil, Anda mengontrol berapa banyak bobot (penekanan) OpenSearch memberikan peringkat dari Amazon Personalize. Melanjutkan contoh ini, jika pengguna mencari jenis mobil tertentu dari tahun tertentu (seperti Toyota Prius 2008), Anda mungkin ingin lebih menekankan pada peringkat asli dari. OpenSearch

Namun, untuk pertanyaan yang lebih umum yang menghasilkan berbagai hasil (seperti pencarian untuk semua kendaraan Toyota), Anda mungkin menempatkan penekanan tinggi pada personalisasi. Dengan cara ini, mobil di bagian atas daftar lebih relevan dengan pengguna tertentu.

Cara kerja plugin Amazon Personalisasi Penelusuran Ranking

Diagram berikut menunjukkan cara kerja plugin Amazon Personalize Search Ranking.



1. Anda mengirimkan kueri pelanggan ke domain OpenSearch Layanan atau OpenSearch kluster open source Anda.
2. OpenSearch mengirimkan respons kueri (daftar item yang relevan dengan kueri) dan ID pengguna ke plugin Amazon Personalize Search Ranking.
3. Plugin mengirimkan item dan pengguna sebagai respons terhadap kampanye Amazon Personalisasi Anda untuk peringkat. Ini menggunakan resep dan kampanye nilai Nama Sumber Daya Amazon (ARN) di pipeline pencarian Anda untuk mendapatkan peringkat yang dipersonalisasi bagi pengguna. Ini menggunakan operasi GetPersonalizedRanking API untuk rekomendasi. Dalam permintaan, ia melewati userId pengguna yang membuat kueri dan item yang dikembalikan dari OpenSearch kueri di. `inputList`
4. Amazon Personalize mengembalikan hasil peringkat ulang ke plugin.
5. Plugin mengatur ulang dan mengembalikan hasil pencarian ke domain OpenSearch Layanan atau kluster open source OpenSearch Anda. Ini memberi peringkat ulang hasil berdasarkan respons dari kampanye Amazon Personalize Anda dan penekanan pada personalisasi yang Anda tentukan selama penyiapan.
6. OpenSearch Cluster open source atau domain OpenSearch Service mengembalikan hasil akhir ke aplikasi Anda.

Informasi tambahan

Sumber daya berikut memberikan informasi tambahan tentang penggunaan OpenSearch.

- Untuk informasi tentang memulai dengan open source OpenSearch, lihat Mulai cepat.
- Untuk informasi tentang memulai OpenSearch Layanan, lihat Memulai OpenSearch Layanan Amazon di Panduan Pengembang OpenSearch Layanan Amazon.

- Untuk informasi tentang resep Personalisasi Peringkat di Amazon Personalisasi, lihat atau. Resep peringkat-v2 yang dipersonalisasi Resep Peringkat yang Dipersonalisasi

Amazon Personalisasi Persyaratan plugin Peringkat Pencarian

Sebelum Anda mulai mempersonalisasi hasil OpenSearch, perhatikan panduan dan persyaratan berikut untuk plugin Amazon Personalize Search Ranking:

- Anda harus menggunakan OpenSearch versi 2.9.0 atau yang lebih baru. Jika Anda menggunakan OpenSearch Layanan Amazon, domain Anda harus menggunakan versi 2.9 atau yang lebih baru.
- Jika Anda belum melakukannya, lengkapi petunjuk Menyiapkan izin untuk memberikan izin kepada pengguna Anda untuk mengakses Amazon Personalisasi dan berikan izin Amazon Personalisasi untuk mengakses sumber daya Anda di Amazon Personalisasi.
- Anda harus dapat mengakses sumber daya Amazon Personalisasi dari domain OpenSearch Layanan atau kluster sumber OpenSearch terbuka.
  - Untuk informasi tentang pemberian akses untuk domain OpenSearch Layanan, lihatMenyiapkan izin OpenSearch Layanan Amazon.
  - Untuk informasi tentang pemberian akses untuk OpenSearch klaster, lihatMenyiapkan OpenSearch izin sumber terbuka.
- Anda hanya dapat menggunakan sumber daya Amazon Personalisasi khusus. Jika Anda membuat grup kumpulan data Domain, Anda masih dapat menambahkan sumber daya khusus.
- Anda hanya dapat menggunakan resep khusus Resep peringkat-v2 yang dipersonalisasi atauResep Peringkat yang Dipersonalisasi.
- Anda harus membuat kumpulan data interaksi Item di Amazon Personalize. Item dan kumpulan data Pengguna bersifat opsional.
- Anda tidak dapat menerapkan filter Amazon Personalize saat menggunakan plugin Amazon Personalize Search Ranking.
- Secara default, plugin mengasumsikan bahwa `_id` untuk dokumen yang diindeks cocok dengan ItemID di OpenSearch Amazon Personalisasi data Anda. Jika OpenSearch data Anda menggunakan bidang berbeda yang sesuai dengan Amazon Personalize ItemIds, Anda harus menentukan nama bidang saat mengonfigurasi plugin.
- UserID yang Anda gunakan untuk pengguna yang membuat kueri harus cocok dengan userId mereka dalam data yang Anda impor ke Amazon Personalize.

- Plugin ini memberi peringkat ulang hanya 500 hasil pencarian teratas dari OpenSearch. Item yang tersisa tidak diberi peringkat ulang dan berakhir di bagian bawah daftar.

Personalisasi hasil dari OpenSearch Layanan Amazon dengan Amazon Personalize

Untuk mempersonalisasi OpenSearch hasil dari Amazon OpenSearch Service, Anda melakukan hal berikut:

1. Siapkan Amazon Personalize — Jika Anda belum melakukannya, selesaikan langkah-langkah Menyiapkan Amazon Personalisasi untuk menyiapkan kredensi dan menyiapkan izin untuk Amazon Personalize. Anda tidak perlu mengatur untuk AWS SDKs mempersonalisasi OpenSearch hasil.
2. Selesaikan alur kerja Amazon Personalize — Lengkapi alur kerja Amazon Personalize untuk mengimpor data, membuat solusi dengan resep Personalized-Ranking-v 2 atau Personalisasi Peringkat, latih versi solusi kustom, dan terapkan dalam kampanye. Anda harus membuat kumpulan data interaksi Item. Dataset Pengguna dan kumpulan data Item bersifat opsional. Untuk informasi selengkapnya, lihat Amazon Personalisasi detail alur kerja.
3. Siapkan izin — Siapkan izin agar Anda dapat mengakses sumber daya Amazon Personalisasi dari OpenSearch domain Layanan. Untuk informasi selengkapnya, lihat Menyiapkan izin.
4. Instal plugin Amazon Personalize Search Ranking - Plugin ini menangani komunikasi dengan Amazon Personalisasi dan peringkat ulang hasil. Untuk informasi tentang menginstal plugin pada domain OpenSearch Layanan, lihatMenginstal plugin.
5. Konfigurasikan plugin Amazon Personalize Search Ranking — Untuk mengonfigurasi plugin, Anda membuat pipeline pencarian. Saluran pipa pencarian adalah kumpulan prosesor permintaan dan respons. Saat membuat pipeline untuk plugin, Anda menentukan sumber daya Amazon Personalisasi dalam prosesor `personalized_search_ranking` respons. Anda juga mengonfigurasi berapa berat plugin yang diberikan hasil dari Amazon Personalisasi saat peringkat ulang hasilnya. Untuk informasi selengkapnya, lihat Membuat pipa.
6. Menerapkan plugin Amazon Personalize Search Ranking ke OpenSearch kueri — Setelah mengonfigurasi pipeline penelusuran dengan prosesor `personalized_search_ranking` respons, Anda siap menerapkan plugin Amazon Personalize Search Ranking ke OpenSearch kueri dan melihat hasil peringkat ulang. Untuk informasi tentang menerapkan plugin ke kueri OpenSearch Layanan, lihatMenerapkan plugin.

7. Bandingkan hasil - Plugin Amazon Personalize Search Ranking memberi peringkat ulang hasil pencarian dalam respons kueri. OpenSearch Ini mempertimbangkan peringkat dari Amazon Personalize dan peringkat dari. OpenSearch Untuk memahami bagaimana hasil diberi peringkat ulang, Anda dapat membandingkan hasil dari kueri yang menggunakan personalisasi dan yang tidak. Untuk informasi tentang membandingkan hasil dengan OpenSearch Layanan, lihatMembandingkan hasil.
8. Pantau plugin Amazon Personalize Search Ranking — Saat Anda menerapkan plugin Amazon Personalize Search Ranking ke kueri penelusuran, Anda dapat memantau plugin dengan mendapatkan metrik untuk pipeline pencarian Anda. Untuk informasi tentang memantau plugin dengan OpenSearch Layanan, lihatMemantau plugin.

Topik

- Menyiapkan izin OpenSearch Layanan Amazon
- Menginstal plugin Amazon Personalize Search Ranking di domain Layanan OpenSearch
- Membuat pipeline di Amazon OpenSearch Service
- Menerapkan plugin ke kueri OpenSearch Layanan Amazon
- Membandingkan hasil OpenSearch Layanan Amazon yang dipersonalisasi dengan hasil tanpa personalisasi
- Memantau plugin dengan Amazon OpenSearch Service

Menyiapkan izin OpenSearch Layanan Amazon

Jika Anda menggunakan OpenSearch Layanan Amazon, Anda harus dapat mengakses sumber daya Amazon Personalisasi dari domain OpenSearch Layanan Anda.

Untuk mengatur izin

1. Bergantung pada apakah sumber daya Anda berada di akun yang sama atau berbeda, buat satu atau beberapa peran layanan IAM dengan izin untuk mengakses sumber daya Anda.
   - Jika sumber daya OpenSearch Layanan dan Amazon Personalisasi berada di akun yang sama, Anda membuat peran layanan IAM untuk OpenSearch Layanan dan memberinya izin untuk mendapatkan peringkat yang dipersonalisasi dari kampanye Amazon Personalisasi. Untuk informasi selengkapnya, lihat Mengkonfigurasi izin saat sumber daya berada di akun yang sama.

- Jika sumber daya OpenSearch Layanan dan Amazon Personalisasi berada di akun terpisah, Anda membuat dua peran layanan IAM. Anda membuatnya di akun dengan sumber daya OpenSearch Layanan Anda dan memberikannya akses ke sumber daya OpenSearch Layanan Anda. Dan Anda membuatnya di akun dengan sumber daya Amazon Personalisasi dan memberinya izin untuk mendapatkan peringkat yang dipersonalisasi dari kampanye Amazon Personalisasi Anda. Untuk informasi selengkapnya, lihat Mengkonfigurasi izin saat sumber daya berada di akun yang berbeda.

2. Berikan pengguna atau peran yang mengakses `PassRole` izin domain OpenSearch Layanan Anda untuk peran layanan IAM yang Anda buat untuk Layanan. OpenSearch Untuk informasi selengkapnya, lihat Mengkonfigurasi keamanan domain OpenSearch Layanan Amazon.

Setelah Anda mengatur izin, Anda siap untuk menginstal plugin di domain Anda. Untuk informasi selengkapnya, lihat Menginstal plugin.

Topik

- Mengkonfigurasi izin saat sumber daya berada di akun yang sama
- Mengkonfigurasi izin saat sumber daya berada di akun yang berbeda
- Mengkonfigurasi keamanan domain OpenSearch Layanan Amazon

Mengkonfigurasi izin saat sumber daya berada di akun yang sama

Jika sumber daya OpenSearch Layanan dan Amazon Personalisasi berada di akun yang sama, Anda harus membuat peran layanan IAM untuk Layanan. OpenSearch Peran ini harus memiliki izin untuk mendapatkan peringkat yang dipersonalisasi dari kampanye Amazon Personalize Anda. Berikut ini diperlukan untuk memberikan izin peran OpenSearch layanan Layanan Anda untuk mendapatkan peringkat yang dipersonalisasi dari kampanye Amazon Personalize Anda:

- Kebijakan kepercayaan peran harus memberikan `AssumeRole` izin untuk OpenSearch Layanan. Untuk contoh kebijakan kepercayaan, lihatContoh kebijakan kepercayaan.
- Peran tersebut harus memiliki izin untuk mendapatkan peringkat yang dipersonalisasi dari kampanye Amazon Personalize Anda. Untuk contoh kebijakan, lihatContoh kebijakan izin.

Untuk informasi tentang membuat peran IAM, lihat Membuat peran IAM di Panduan Pengguna IAM. Untuk informasi tentang melampirkan kebijakan IAM ke peran, lihat Menambahkan dan menghapus izin identitas IAM di Panduan Pengguna IAM.

Setelah membuat peran layanan IAM untuk OpenSearch Layanan, Anda harus memberikan pengguna atau peran yang mengakses `PassRole` izin domain OpenSearch Layanan untuk peran OpenSearch Layanan. Untuk informasi selengkapnya, lihat Mengkonfigurasi keamanan domain OpenSearch Layanan Amazon.

Topik

- Contoh kebijakan kepercayaan
- Contoh kebijakan izin

Contoh kebijakan kepercayaan

Contoh kebijakan kepercayaan berikut memberikan `AssumeRole` izin untuk OpenSearch Layanan.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "",
        "Effect": "Allow",
        "Action": "sts:AssumeRole",
        "Principal": {
            "Service": [
                "es.amazonaws.com"
            ]
        }
    }]
}
```

Contoh kebijakan izin

Contoh kebijakan berikut memberi peran izin minimum untuk mendapatkan peringkat yang dipersonalisasi dari kampanye Amazon Personalize Anda. Untuk`Campaign ARN`, tentukan Nama Sumber Daya Amazon (ARN) kampanye Amazon Personalize Anda.

JSON

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "personalize:GetPersonalizedRanking"
            ],
            "Resource": "arn:aws:personalize:us-
east-1:111122223333:campaign/YourResourceId"
        }
    ]
}
```

Mengkonfigurasi izin saat sumber daya berada di akun yang berbeda

Jika OpenSearch Layanan dan Amazon Personalisasi sumber daya berada di akun terpisah, Anda membuat peran IAM di setiap akun dan memberikan akses peran ke sumber daya di akun.

Untuk mengatur izin untuk beberapa akun

1. Di akun tempat kampanye Amazon Personalisasi Anda ada, buat peran IAM yang memiliki izin untuk mendapatkan peringkat yang dipersonalisasi dari kampanye Amazon Personalize Anda. Saat Anda mengkonfigurasi plugin, Anda menentukan ARN untuk peran ini dalam `external_account_iam_role_arn` parameter prosesor `personalized_search_ranking` respons. Untuk informasi selengkapnya, lihat Membuat pipeline di Amazon OpenSearch Service.

   Untuk contoh kebijakan, lihatContoh kebijakan izin.
2. Di akun tempat domain OpenSearch Layanan Anda ada, buat peran dengan kebijakan kepercayaan yang memberikan `AssumeRole` izin OpenSearch Layanan. Saat Anda mengkonfigurasi plugin, Anda menentukan ARN untuk peran ini dalam `iam_role_arn` parameter prosesor `personalized_search_ranking` respons. Untuk informasi selengkapnya, lihat Membuat pipeline di Amazon OpenSearch Service.

   Untuk contoh kebijakan kepercayaan, lihatContoh kebijakan kepercayaan.
3. Ubah setiap peran untuk memberikan `AssumeRole` izin peran lainnya. Misalnya, untuk peran yang memiliki akses ke sumber daya Amazon Personalisasi Anda, kebijakan IAM akan memberikan peran dalam akun dengan domain OpenSearch Layanan mengambil izin peran sebagai berikut:

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "",
        "Effect": "Allow",
        "Action": "sts:AssumeRole",
        "Resource": "arn:aws:iam::111122223333:role/roleName"

    }]
}
```

4. Di akun tempat domain OpenSearch Layanan Anda ada, berikan pengguna atau peran yang mengakses `PassRole` izin domain OpenSearch Layanan Anda untuk peran layanan OpenSearch Layanan yang baru saja Anda buat. Untuk informasi selengkapnya, lihat Mengkonfigurasi keamanan domain OpenSearch Layanan Amazon.

Mengkonfigurasi keamanan domain OpenSearch Layanan Amazon

Untuk menggunakan plugin dengan OpenSearch Layanan, pengguna atau peran yang mengakses domain Anda harus memiliki `PassRole` izin untuk peran layanan IAM untuk Layanan yang baru saja Anda buat OpenSearch . Juga, pengguna atau peran harus memiliki izin untuk melakukan `es:ESHttpGet` dan `es:ESHttpPut` tindakan.

Untuk informasi tentang mengonfigurasi akses ke OpenSearch Layanan, lihat Keamanan di OpenSearch Layanan Amazon di Panduan Pengembang OpenSearch Layanan Amazon. Untuk contoh kebijakan, lihatContoh kebijakan untuk pengguna atau peran OpenSearch Layanan.

Contoh kebijakan untuk pengguna atau peran OpenSearch Layanan

Contoh kebijakan IAM berikut memberikan `PassRole` izin pengguna atau peran untuk peran layanan IAM yang Anda buat untuk Layanan. OpenSearch Mengkonfigurasi izin saat sumber daya berada di akun yang sama

JSON

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/aws-service-role/
opensearchservice.amazonaws.com/AWSServiceRoleForAmazonOpenSearchService"
        }
    ]
}
```

Kebijakan IAM berikut memberikan izin minimum untuk membuat pipeline dan kueri penelusuran dengan Service. OpenSearch

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "es:ESHttpGet",
                "es:ESHttpPut"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:ResourceTag/environment": [
                        "production"
                    ]
                }
            }
        }
    ]
}
```

Menginstal plugin Amazon Personalize Search Ranking di domain Layanan OpenSearch

Setelah menyelesaikan alur kerja Amazon Personalize dan memenuhi persyaratan yang tercantumPersyaratan plugin, Anda siap untuk menginstal plugin di domain Anda.

Untuk menggunakan plugin, Anda mengaitkan `Amazon_Personalize_Search_Ranking_Plugin` plugin dengan domain Anda. Plugin sudah diinstal sebelumnya, dan Anda tidak perlu mengimpornya dari Amazon S3. Anda mengaitkan plugin dengan cara yang sama seperti Anda mengaitkan paket OpenSearch Layanan. Untuk informasi tentang mengaitkan paket OpenSearch Layanan, lihat Paket khusus untuk OpenSearch Layanan Amazon.

Setelah Anda mengaitkan plugin dengan domain Anda, Anda siap untuk mengkonfigurasi plugin. Anda mengonfigurasinya dengan membuat pipeline pencarian dan menentukan prosesor `personalized_search_ranking` respons. Untuk informasi selengkapnya, lihat Membuat pipa.

Informasi tambahan tentang domain OpenSearch Layanan Amazon

Sumber daya berikut memberikan informasi tambahan tentang penggunaan domain OpenSearch Layanan Amazon.

- Untuk tutorial singkat untuk mengonfigurasi domain pengujian, lihat Langkah 1: Membuat domain OpenSearch Layanan Amazon di bagian “Memulai” dari Panduan Pengembang OpenSearch Layanan Amazon.
- Untuk langkah selengkapnya tentang mengonfigurasi domain OpenSearch Layanan, lihat Membuat dan mengelola domain OpenSearch Layanan Amazon.
- Untuk tutorial singkat tentang mengunggah sejumlah kecil data pengujian ke OpenSearch Layanan, lihat Langkah 2: Mengunggah data ke OpenSearch Layanan Amazon untuk diindeks di bagian “Memulai” pada Panduan Pengembang OpenSearch Layanan Amazon.
- Untuk informasi selengkapnya tentang menelan data, lihat Mengindeks data di OpenSearch Layanan Amazon di Panduan Pengembang OpenSearch Layanan Amazon.

Membuat pipeline di Amazon OpenSearch Service

Setelah menginstal plugin Amazon Personalize Search Ranking, Anda siap mengonfigurasinya dengan membuat pipeline OpenSearch pencarian.

Pipeline pencarian adalah sekumpulan prosesor permintaan dan respons yang berjalan secara berurutan sesuai urutan yang Anda buat. Saat Anda membuat pipeline pencarian untuk plugin, Anda menentukan prosesor `personalized_search_ranking` respons. Untuk informasi tentang saluran pipa pencarian, lihat Cari saluran pipa.

Setelah membuat pipeline pencarian dengan prosesor `personalized_search_ranking` respons, Anda siap untuk mulai menerapkan plugin ke OpenSearch kueri. Anda dapat menerapkannya ke OpenSearch indeks atau OpenSearch kueri individual. Untuk informasi selengkapnya, lihat Menerapkan plugin.

Anda dapat menggunakan kode Python berikut untuk membuat pipeline pencarian dengan prosesor `personalized_search_ranking` respons pada domain OpenSearch Layanan. Ganti `domain endpoint` dengan URL endpoint domain Anda. Sebagai contoh: `https://<domain name>.<AWS region>.es-staging.amazonaws.com`. Untuk penjelasan lengkap dari setiap `personalized_search_ranking` parameter, lihatBidang untuk prosesor `personalized_search_ranking` respons.

```
import requests
from requests_auth_aws_sigv4 import AWSSigV4

domain_endpoint = 'domain endpoint'
pipeline_name = 'pipeline name'
url = f'{domain_endpoint}/_search/pipeline/{pipeline_name}'
auth = AWSSigV4('es')

headers = {'Content-Type': 'application/json'}

body = {
  "description": "A pipeline to apply custom re-ranking from Amazon Personalize",
  "response_processors": [
    {
      "personalized_search_ranking" : {
        "campaign_arn" : "Amazon Personalize Campaign ARN",
        "item_id_field" : "productId",
        "recipe" : "aws-personalized-ranking-v2",
        "weight" : "0.3",
        "tag" : "personalize-processor",
        "iam_role_arn": "Role ARN",
        "aws_region": "AWS region",
        "ignore_failure": true
    }
  ]
```

```
}
try:
    response = requests.put(url, auth=auth, json=body, headers=headers, verify=False)
    print(response.text)
except Exception as e:
    print(f"Error: {e}")
```

Menerapkan plugin ke kueri OpenSearch Layanan Amazon

Setelah membuat pipeline, Anda siap menerapkan plugin Amazon Personalize Search Ranking ke kueri. Anda dapat menerapkan plugin Amazon Personalize Search Ranking ke semua kueri dan tanggapan untuk indeks. Anda juga dapat menerapkan plugin untuk pertanyaan dan tanggapan individual.

- Anda dapat menggunakan kode Python berikut untuk menerapkan pipeline pencarian ke indeks. Dengan pendekatan ini, semua pencarian menggunakan indeks ini menggunakan plugin untuk menerapkan personalisasi ke hasil pencarian.

  ```
  import requests
  from requests_auth_aws_sigv4 import AWSSigV4

  domain_endpoint = 'domain endpoint'
  index = 'index name'
  url = f'{domain_endpoint}/{index}/_settings/'
  auth = AWSSigV4('es')
  headers = {'Content-Type': 'application/json'}
  body = {
      "index.search.default_pipeline": "pipeline name"
  }
  try:
      response = requests.put(url, auth=auth, json=body, headers=headers)
      print(response.text)
  except Exception as e:
      print(f"Error: {e}")
  ```

- Anda dapat menggunakan kode Python berikut untuk menerapkan pipeline pencarian ke kueri individual untuk mobil merek Toyota.

  Perbarui kode untuk menentukan titik akhir domain, indeks OpenSearch Layanan, nama pipeline, dan kueri. Untuk`user_id`, tentukan ID pengguna tempat Anda mendapatkan hasil penelusuran. Pengguna ini harus berada dalam data yang Anda gunakan untuk membuat versi solusi Amazon

Personalize Anda. Jika pengguna tidak hadir, Amazon Personalize memberi peringkat item berdasarkan popularitasnya.

Untuk`context`, jika Anda menggunakan metadata kontekstual, berikan metadata kontekstual pengguna, seperti jenis perangkat mereka. `context`Bidang ini opsional. Lihat informasi yang lebih lengkap di Meningkatkan relevansi rekomendasi dengan metadata kontekstual.

```
import requests
from requests_auth_aws_sigv4 import AWSSigV4

domain_endpoint = 'domain endpoint'
index = 'index name'
url = f'{domain_endpoint}/{index}/_search/'

auth = AWSSigV4('es')
headers = {'Content-Type': 'application/json'}
params = {"search_pipeline": "pipeline-name"}
body = {
    "query": {
        "multi_match": {
            "query": "Toyota",
            "fields": ["BRAND"]
        }
    },
    "ext": {
        "personalize_request_parameters": {
            "user_id": "USER ID",
            "context": { "DEVICE" : "mobile phone" }
        }
    }
}
try:
    response = requests.post(url, auth=auth, params=params, json=body,
 headers=headers, verify=False)
    print(response)
except Exception as e:
    print(f"Error: {e}")
```

Membandingkan hasil OpenSearch Layanan Amazon yang dipersonalisasi dengan hasil tanpa personalisasi

Plugin Amazon Personalize Search Ranking mengatur ulang hasil pencarian berdasarkan peringkat dari Amazon Personalize dan peringkat dari. OpenSearch Cara plugin memberi peringkat ulang hasil tergantung pada bagaimana Anda mengonfigurasi prosesor `personalized_search_ranking` respons di saluran pipa Anda.

Untuk memahami bagaimana hasil diberi peringkat, Anda dapat menjalankan kueri dengan dan tanpa personalisasi, dan membandingkan hasilnya. Anda dapat menggunakan kode Python berikut untuk menjalankan dua query yang berbeda dan output hasilnya ke dua file JSON. Metode pertama menjalankan kueri yang menggunakan plugin untuk menentukan peringkat ulang hasil. Yang kedua menjalankan metode yang menghasilkan hasil tanpa personalisasi.

```
import json
import requests
from requests_auth_aws_sigv4 import AWSSigV4


# Returns re-ranked OpenSearch results using the Amazon Personalize Search Ranking
 plugin.
def get_personalized_results(pipeline_name):
    url = f'{domain}/{index}/_search/'
    auth = AWSSigV4('es')
    headers = {'Content-Type': 'application/json'}
    params = {"search_pipeline": pipeline_name}
    body = {
        "query": {
            "multi_match": {
                "query": "Toyota",
                "fields": ["BRAND"]
            }
        },
        "ext": {
            "personalize_request_parameters": {
                "user_id": "1"
            }
        }
    }
    try:
```

```
        response = requests.post(url, auth=auth, params=params, json=body,
 headers=headers, verify=False)
    except Exception as e:
        return f"Error: {e}"
    return response.text


# Returns OpenSearch results without personalization.
def get_opensearch_results():
    url = f'{domain}/{index}/_search/'
    auth = AWSSigV4('es')
    headers = {'Content-Type': 'application/json'}
    body = {
        "query": {
            "multi_match": {
                "query": "Toyota",
                "fields": ["BRAND"]
            }
        }
    }
    try:
        response = requests.post(url, auth=auth, json=body, headers=headers,
 verify=False)
    except Exception as e:
        return f"Error: {e}"
    return response.text


def print_results(file_name, results):
    results_file = open(file_name, 'w')
    results_file.write(json.dumps(results, indent=4))
    results_file.close()


# specify domain endpoint
domain = "DOMAIN_ENDPOINT"

# specify the region where you created your Amazon Personalize resources and Amazon
 OpenSearch domain
aws_region = "REGION"

# specify the name of the pipeline that uses the Amazon Personalize plugin
pipeline_name = "PIPELINE_NAME"
```

```
# specify your Amazon OpenSearch index
index = "INDEX"

# specify names for json files for comparison
personalized_results_file = "personalized_results.json"
opensearch_results_file = "opensearch_results.json"

# get personalized results
personalized_results = json.loads(get_personalized_results(pipeline_name))

# get OpenSearch results without personalization
opensearch_results = json.loads(get_opensearch_results())

# print results to files
print_results(personalized_results_file, personalized_results)
print_results(opensearch_results_file, opensearch_results)
```

Memantau plugin dengan Amazon OpenSearch Service

Saat Anda menerapkan plugin Amazon Personalize Search Ranking ke OpenSearch kueri, Anda dapat memantau plugin dengan mendapatkan metrik untuk pipeline pencarian Anda. Metrik pipeline mencakup statistik seperti jumlah permintaan gagal untuk prosesor `personalized_search_ranking` respons.

Jika Anda menggunakan OpenSearch Layanan, Anda dapat memantau plugin melalui metrik di Amazon CloudWatch. Untuk informasi selengkapnya, lihat Memantau domain OpenSearch Layanan Amazon.

Anda dapat menggunakan kode Python berikut untuk mendapatkan metrik untuk semua pipeline Anda. Untuk contoh metrik pipeline, lihatContoh metrik pipa.

```
import requests
from requests_auth_aws_sigv4 import AWSSigV4

domain_endpoint = 'domain endpoint'
url = f'{domain_endpoint}/_nodes/stats/search_pipeline'

auth = AWSSigV4('es')
headers = {'Content-Type': 'application/json'}
try:
    response = requests.get(url, auth=auth, headers=headers, verify=False)
    print(response.text)
```

```
except Exception as e:
    print(f"Error: {e}")
```

Mempersonalisasi hasil dari Open Search open source dengan Amazon Personalize

Untuk mempersonalisasi hasil open source OpenSearch, Anda melakukan hal berikut:

1. Siapkan Amazon Personalize — Jika Anda belum melakukannya, selesaikan langkah-langkah Menyiapkan Amazon Personalisasi untuk menyiapkan kredensi dan menyiapkan izin untuk Amazon Personalize. Anda tidak perlu mengatur untuk AWS SDKs mempersonalisasi OpenSearch hasil.
2. Selesaikan alur kerja Amazon Personalize — Lengkapi alur kerja Amazon Personalize untuk mengimpor data, membuat solusi dengan resep resep Personalized-Ranking-v 2 atau Personalisasi Peringkat, latih versi solusi kustom, dan terapkan dalam kampanye. Anda harus membuat kumpulan data interaksi Item. Dataset Pengguna dan kumpulan data Item bersifat opsional. Untuk informasi selengkapnya, lihat Amazon Personalisasi detail alur kerja.
3. Siapkan izin — Siapkan izin agar Anda dapat mengakses sumber daya Amazon Personalisasi dari klaster. OpenSearch Untuk informasi selengkapnya, lihat Menyiapkan izin.
4. Instal plugin Amazon Personalize Search Ranking - Plugin Amazon Personalize Search Ranking menangani komunikasi dengan Amazon Personalisasi dan peringkat ulang hasil.
   - Jika Anda sudah menjalankan OpenSearch cluster, Anda dapat menginstal plugin secara manual. Untuk informasi selengkapnya, lihat Menginstal plugin secara manual pada OpenSearch cluster yang ada.
   - Jika Anda belum membuat OpenSearch cluster, Anda dapat menggunakan skrip bash quickstart untuk membuatnya. Untuk informasi selengkapnya, lihat Membuat cluster baru dan menginstal plugin dengan skrip.
5. Konfigurasikan plugin Amazon Personalize Search Ranking — Untuk mengonfigurasi plugin, Anda membuat pipeline pencarian. Saluran pipa pencarian adalah kumpulan prosesor permintaan dan respons. Saat membuat pipeline untuk plugin, Anda menentukan sumber daya Amazon Personalisasi dalam prosesor `personalized_search_ranking` respons. Anda juga mengonfigurasi berapa berat plugin yang diberikan hasil dari Amazon Personalisasi saat peringkat ulang hasilnya. Untuk informasi selengkapnya, lihat Membuat pipa.
6. Terapkan plugin Amazon Personalize Search Ranking ke OpenSearch kueri - Anda dapat menerapkan plugin Amazon Personalize Search Ranking ke semua kueri dan tanggapan untuk

indeks. OpenSearch Anda juga dapat menerapkan plugin untuk OpenSearch pertanyaan dan tanggapan individual. Untuk informasi tentang menerapkan plugin ke kueri di open source OpenSearch, lihatMenerapkan plugin.

7. Bandingkan hasil - Plugin Amazon Personalize Search Ranking memberi peringkat ulang hasil pencarian dalam respons kueri. OpenSearch Ini mempertimbangkan peringkat dari Amazon Personalize dan peringkat dari. OpenSearch Untuk memahami bagaimana hasil diberi peringkat ulang, Anda dapat membandingkan hasil dari kueri yang menggunakan personalisasi dan yang tidak. Untuk informasi tentang membandingkan hasil dengan open source OpenSearch, lihatMembandingkan hasil.
8. Pantau plugin Amazon Personalize Search Ranking — Saat Anda menerapkan plugin Amazon Personalize Search Ranking ke kueri penelusuran, Anda dapat memantau plugin dengan mendapatkan metrik untuk pipeline pencarian Anda. Untuk informasi tentang pemantauan plugin pada OpenSearch cluster open source, lihatMemantau plugin dengan open source OpenSearch. Untuk kutipan metrik pipeline yang dikembalikan, lihat. OpenSearch Contoh metrik pipa

Topik

- Menyiapkan OpenSearch izin sumber terbuka
- Menginstal plugin Amazon Personalize Search Ranking secara manual di cluster yang ada OpenSearch
- Membuat cluster baru dan menginstal plugin dengan skrip
- Membuat pipeline di open source OpenSearch
- Menerapkan plugin Amazon Personalize Search Ranking ke kueri di open source OpenSearch
- Membandingkan OpenSearch hasil yang dipersonalisasi dengan hasil tanpa personalisasi
- Memantau plugin dengan open source OpenSearch

Menyiapkan OpenSearch izin sumber terbuka

Jika Anda menggunakan open source OpenSearch, Anda harus dapat mengakses sumber daya Amazon Personalize dari kluster penelusuran terbuka Anda. Untuk memberikan akses, lakukan hal berikut:

- Jika Anda menyiapkan OpenSearch dari awal, Anda dapat menggunakan skrip bash mulai cepat untuk menjalankan OpenSearch cluster di wadah Docker. Skrip menggunakan kredensi default di profil Anda AWS . Anda dapat menentukan profil alternatif saat menjalankan skrip.

Kredensi ini harus dikaitkan dengan pengguna atau peran yang memiliki izin untuk melakukan GetPersonalizedRanking tindakan untuk kampanye Amazon Personalize Anda. Untuk contoh kebijakan IAM, lihatContoh kebijakan IAM. Atau, kredensional harus memiliki izin untuk mengambil peran yang memiliki izin ini. Anda dapat memberikan Amazon Resource Name (ARN) untuk peran ini saat membuat pipeline untuk plugin Amazon Personalize Search Ranking.

- Jika Anda tidak menggunakan skrip bash mulai cepat, Anda dapat menambahkan kredensional Anda secara manual ke keystore Anda. OpenSearch Kredensi ini harus sesuai dengan pengguna atau peran yang memiliki izin untuk melakukan GetPersonalizedRanking tindakan untuk kampanye Amazon Personalize Anda.

  Untuk menambahkan AWS kredenal Anda secara manual ke OpenSearch keystore Anda, jalankan perintah berikut di mana OpenSearch cluster Anda berjalan (seperti wadah Docker). Kemudian berikan setiap kredensi. Jika Anda tidak menggunakan token sesi, Anda dapat menghilangkan baris terakhir dalam perintah.

  ```
  opensearch-keystore add \
  personalized_search_ranking.aws.access_key \
  personalized_search_ranking.aws.secret_key \
  personalized_search_ranking.aws.session_token
  ```

- Jika menjalankan OpenSearch klaster di EC2 instans Amazon, Anda dapat memberikan izin dengan profil instans IAM. Kebijakan yang dilampirkan pada peran harus memberinya izin untuk melakukan GetPersonalizedRanking tindakan untuk kampanye Amazon Personalisasi Anda. Itu juga harus memberikan EC2 izin Amazon untuk mengambil peran.

  Untuk informasi tentang profil EC2 instans Amazon, lihat Menggunakan profil instans. Untuk contoh kebijakan, lihatContoh kebijakan IAM.

Contoh kebijakan IAM

Contoh kebijakan berikut memberi pengguna atau peran izin minimum untuk mendapatkan peringkat yang dipersonalisasi dari kampanye Amazon Personalize Anda. Untuk`Campaign ARN`, tentukan Nama Sumber Daya Amazon (ARN) kampanye Amazon Personalize Anda.

JSON

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "personalize:GetPersonalizedRanking"
            ],
            "Resource": "arn:aws:personalize:us-
east-1:111122223333:campaign/YourResourceId"
        }
    ]
}
```

Selain itu, jika Anda menjalankan OpenSearch klaster di EC2 instans Amazon dan memberikan izin dengan profil instans IAM, kebijakan kepercayaan untuk peran tersebut harus memberikan EC2 `AssumeRole` izin Amazon sebagai berikut. Untuk informasi tentang profil EC2 instans Amazon, lihat Menggunakan profil instans.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "ec2.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Menginstal plugin Amazon Personalize Search Ranking secara manual di cluster yang ada OpenSearch

Jika Anda sudah memiliki OpenSearch cluster, Anda dapat menginstal plugin secara manual di cluster Anda langsung dari OpenSearch GitHub repositori.

Untuk menginstal plugin secara manual

1. Gunakan perintah berikut untuk memulai OpenSearch cluster Anda:

   ```
   bin/opensearch
   ```

2. Jika Anda belum melakukannya, unggah data katalog Anda ke OpenSearch klaster Anda. Saat mengunggah data, Anda membuat OpenSearch indeks dan menentukan pemetaan bidang Anda. Kemudian Anda mengunggah data Anda ke indeks itu. Sebagai contoh, lihat Membuat indeks dan pemetaan bidang menggunakan data sampel.
3. Gunakan perintah berikut untuk menginstal plugin:

   ```
   bin/opensearch-plugin install https://github.com/opensearch-project/search-
   processor/releases/download/2.9.0/opensearch-search-processor-2.9.0.0.zip
   ```

   Untuk informasi selengkapnya tentang menginstal plugin, lihat Menginstal plugin.

Setelah Anda menginstal plugin Amazon Personalize Search Ranking, Anda siap untuk mengonfigurasinya. Anda mengonfigurasi plugin dengan membuat pipeline pencarian dan menentukan prosesor `personalized_search_ranking` respons. Lihat informasi yang lebih lengkap di Membuat pipa.

Membuat cluster baru dan menginstal plugin dengan skrip

Jika Anda belum membuat OpenSearch cluster, Anda dapat menggunakan skrip bash quickstart untuk membuatnya. Skrip ini menyiapkan OpenSearch cluster dalam wadah Docker, menyiapkan kredensyal menggunakan AWS profil default Anda, dan menginstal plugin Amazon Personalize Search Ranking.

Untuk informasi tentang membuat OpenSearch cluster secara manual, lihat petunjuk Mulai Cepat di OpenSearch dokumentasi.

Untuk menginstal plugin dengan skrip bash quickstart

1. Sebelum Anda menjalankan skrip, unduh dan instal Docker Desktop untuk sistem operasi Anda.
2. Unduh skrip bash mulai cepat dari GitHub.
3. Di direktori kerja Anda, jalankan skrip dengan perintah berikut.

```
sh personalized_search_ranking_quickstart.sh
```

Dengan perintah ini, skrip menggunakan kredensyal di profil default AWS Anda. Untuk memberikan profil alternatif, gunakan `--profile` argumen.

```
sh personalized_search_ranking_quickstart.sh --profile profile-name
```

Setelah Anda menjalankan skrip, Anda dapat menemukan informasi lebih lanjut tentang skrip dalam file README yang terletak di direktori unik yang dibuat oleh skrip. Direktori ini menyimpan file Dockerfile dan docker-compose.yl yang digunakan skrip. Sebagai contoh: `../opensearch-personalize-intelligent-ranking-docker.1234/README`.

4. Unggah data katalog Anda ke OpenSearch klaster Anda. Saat mengunggah data, Anda membuat OpenSearch indeks dan menentukan pemetaan bidang Anda. Kemudian Anda mengunggah data Anda ke indeks itu. Sebagai contoh, lihat Membuat indeks dan pemetaan bidang menggunakan data sampel.

Setelah menyiapkan OpenSearch dan menginstal plugin Amazon Personalize Search Ranking, Anda siap mengonfigurasinya. Anda mengonfigurasi plugin dengan membuat pipeline pencarian dan menentukan prosesor `personalized_search_ranking` respons. Lihat informasi yang lebih lengkap di Membuat pipa.

Membuat pipeline di open source OpenSearch

Setelah Anda menginstal plugin di cluster Anda, Anda siap untuk mengkonfigurasinya dengan membuat pipeline OpenSearch pencarian.

Pipeline pencarian adalah sekumpulan prosesor permintaan dan respons yang berjalan secara berurutan sesuai urutan yang Anda buat. Saat Anda membuat pipeline pencarian untuk plugin, Anda menentukan prosesor `personalized_search_ranking` respons. Untuk informasi tentang saluran pipa pencarian, lihat Cari saluran pipa.

Setelah Anda membuat pipeline dengan prosesor `personalized_search_ranking` respons, Anda siap untuk mulai menerapkan plugin ke kueri. Untuk informasi selengkapnya, lihat Menerapkan plugin.

Anda dapat menggunakan perintah curl berikut untuk membuat pipeline pencarian dengan prosesor `personalized_search_ranking` respons pada OpenSearch cluster open source. Untuk

penjelasan lengkap dari setiap `personalized_search_ranking` parameter, lihatBidang untuk prosesor `personalized_search_ranking` respons.

```
curl -X PUT "http://localhost:9200/_search/pipeline/pipeline-name" -ku 'admin:admin' --
insecure -H 'Content-Type: application/json' -d'
{
  "description": "A pipeline to apply custom re-ranking from Amazon Personalize",
  "response_processors" : [
    {
      "personalized_search_ranking" : {
        "campaign_arn" : "Amazon Personalize Campaign ARN",
        "item_id_field" : "productId",
        "recipe" : "aws-personalized-ranking-v2",
        "weight" : "0.3",
        "tag" : "personalize-processor",
        "iam_role_arn": "Role ARN",
        "aws_region": "AWS region",
        "ignore_failure": true
      }
    }
  ]
}'
```

Setelah membuat pipeline pencarian dengan prosesor `personalized_search_ranking` respons, Anda siap untuk mulai menerapkan plugin ke OpenSearch kueri. Anda dapat menerapkannya ke OpenSearch indeks atau OpenSearch kueri individual. Lihat informasi yang lebih lengkap di Menerapkan plugin Amazon Personalize Search Ranking ke kueri di open source OpenSearch.

Menerapkan plugin Amazon Personalize Search Ranking ke kueri di open source OpenSearch

Anda dapat menerapkan plugin Amazon Personalize Search Ranking ke semua kueri dan tanggapan untuk indeks. OpenSearch Anda juga dapat menerapkan plugin untuk OpenSearch pertanyaan dan tanggapan individual.

- Perintah curl berikut menerapkan pipeline pencarian ke OpenSearch indeks di OpenSearch cluster open source yang berjalan secara lokal. Dengan pendekatan ini, semua pencarian di indeks ini menggunakan plugin untuk menerapkan personalisasi ke hasil pencarian.

  ```
  curl -XGET "https://localhost:9200/index/_settings" -ku 'admin:admin' --insecure -H
   'Content-Type: application/json' -d'
  ```

```
{
  "index.search.default_pipeline": "pipeline-name"
}
'
```

- Perintah curl berikut menerapkan pipeline pencarian ke kueri individu untuk mobil merek Toyota pada indeks di OpenSearch cluster open source yang berjalan secara lokal.

  Untuk`user_id`, tentukan ID pengguna tempat Anda mendapatkan hasil penelusuran. Pengguna ini harus berada dalam data yang Anda gunakan untuk membuat versi solusi Amazon Personalize Anda. Jika pengguna tidak hadir, Amazon Personalize memberi peringkat item berdasarkan popularitasnya. Untuk`context`, jika Anda menggunakan metadata kontekstual, berikan metadata kontekstual pengguna, seperti jenis perangkat mereka. `context`Bidang ini opsional. Untuk informasi selengkapnya, lihat Meningkatkan relevansi rekomendasi dengan metadata kontekstual.

```
curl -XGET "http://localhost:9200/index/_search?search_pipeline=pipeline-name" -ku
 'admin:admin' --insecure -H 'Content-Type: application/json' -d'
{
  "query": {
    "multi_match": {
      "query": "Toyota",
      "fields": ["BRAND"]
    }
  },
  "ext": {
    "personalize_request_parameters": {
      "user_id": "USER ID",
      "context": { "DEVICE": "mobile phone" }
    }
  }
}
'
```

Untuk memahami bagaimana hasil diberi peringkat ulang, Anda dapat menggunakan OpenSearch Dasbor untuk membandingkan OpenSearch hasil dengan hasil yang diberi peringkat ulang dengan plugin. Untuk informasi selengkapnya, lihat Membandingkan OpenSearch hasil yang dipersonalisasi dengan hasil tanpa personalisasi.

Saat Anda menerapkan plugin ke OpenSearch kueri, Anda dapat memantau plugin dengan mendapatkan metrik untuk pipeline Anda OpenSearch. Lihat informasi yang lebih lengkap di Memantau plugin dengan open source OpenSearch.

Membandingkan OpenSearch hasil yang dipersonalisasi dengan hasil tanpa personalisasi

Untuk memahami bagaimana hasil diberi peringkat ulang, Anda dapat menjalankan kueri dengan konsol Alat Dev di dua jendela browser terpisah. Kemudian Anda dapat membandingkan hasil untuk kueri dengan dan tanpa personalisasi.

Untuk membandingkan hasil dengan konsol Dev Tools

1. Pastikan OpenSearch Dashboard sudah terpasang. Skrip bash quickstart menginstal OpenSearch Dasbor. Jika Anda tidak menggunakan skrip atau sudah menjalankan cluster, Anda harus menginstal OpenSearch Dasbor. Untuk informasi selengkapnya, lihat Menginstal OpenSearch Dasbor.
2. Luncurkan OpenSearch Dasbor. Buka `http://localhost:5601` dari browser dan masuk ke OpenSearch Dasbor. Kredensi default adalah nama pengguna 'admin' dan kata sandi 'admin'.
3. Pilih Alat Pengembang di bawah menu Manajemen di halaman beranda OpenSearch Dasbor.
4. Buka jendela browser terpisah dan buka konsol Dev Tools lagi. Anda dapat menggunakan URL dari jendela sebelumnya.
5. Dalam satu jendela, masukkan kueri yang tidak menggunakan peringkat ulang apa pun untuk personalisasi. Di jendela lain, masukkan perintah curl yang menggunakan pipeline dengan prosesor `personalized_search_ranking` respons. Jika Anda menempelkan perintah curl langsung ke konsol, perintah tersebut secara otomatis diubah menjadi format yang digunakan konsol. Untuk contoh perintah, lihatMenerapkan plugin Amazon Personalize Search Ranking ke kueri di open source OpenSearch.
6. Jalankan kedua kueri dan bandingkan hasilnya.

Memantau plugin dengan open source OpenSearch

Saat Anda menerapkan plugin Amazon Personalize Search Ranking ke OpenSearch kueri, Anda dapat memantau plugin dengan mendapatkan metrik untuk pipeline pencarian Anda. Metrik pipeline mencakup statistik seperti jumlah permintaan gagal untuk prosesor `personalized_search_ranking` respons.

Anda dapat menggunakan kode berikut untuk mendapatkan metrik untuk semua pipeline Anda. Respons berisi statistik untuk semua pipeline pencarian. Untuk contoh metrik pipeline, lihatContoh metrik pipa.

```
curl -XGET "https://localhost:9200/_nodes/stats/search_pipeline?pretty" -ku
 'admin:admin'
```

Bidang untuk prosesor **`personalized_search_ranking`** respons

Saat membuat pipeline penelusuran untuk plugin Amazon Personalize Search Ranking, Anda menentukan prosesor `personalized_search_ranking` respons dengan bidang berikut.

- campaign_arn (wajib) - Tentukan Nama Sumber Daya Amazon (ARN) dari kampanye Amazon Personalize yang akan digunakan untuk mempersonalisasi hasil.
- item_id_field (opsional) - Jika `_id` bidang untuk dokumen yang diindeks tidak OpenSearch sesuai dengan Amazon Personalize ItemIds Anda, tentukan nama bidang yang melakukannya. Secara default, plugin mengasumsikan bahwa `_id` data cocok dengan ItemID di Amazon Personalisasi data Anda.
- resep (wajib) — Tentukan nama resep Amazon Personalize yang akan digunakan. Untuk menggunakan plugin, Anda dapat menentukan `aws-personalized-ranking-v2` atau`aws-personalized-ranking`.
- bobot (wajib) - Tentukan penekanan yang diberikan prosesor respons pada personalisasi saat memberi peringkat ulang hasil. Tentukan nilai dalam kisaran 0,0—1,0. Semakin dekat dengan `1.0` itu, semakin besar kemungkinan hasil dari Amazon Personalisasi peringkat lebih tinggi. Jika Anda menentukan`0.0`, tidak ada personalisasi yang terjadi dan OpenSearch diutamakan.
- tag (opsional) - Tentukan pengenal untuk prosesor.
- iam_role_arn (diperlukan untuk OpenSearch Layanan, opsional untuk sumber terbuka OpenSearch) — Untuk Layanan OpenSearch , berikan Nama Sumber Daya Amazon (ARN) untuk peran yang Anda buat saat menyiapkan izin untuk Layanan untuk mengakses sumber daya Amazon Personalisasi. OpenSearch Jika OpenSearch Layanan dan Amazon Personalisasi sumber daya Anda ada di akun yang berbeda, tentukan peran yang memberikan `AssumeRole` izin untuk Layanan. OpenSearch Untuk informasi selengkapnya, lihat Mengkonfigurasi izin saat sumber daya berada di akun yang berbeda.

Untuk open source OpenSearch, jika Anda menggunakan beberapa peran untuk membatasi izin untuk grup pengguna yang berbeda di organisasi Anda, tentukan ARN peran yang memiliki izin untuk mengakses Amazon Personalize. Jika Anda hanya menggunakan AWS kredensyal di OpenSearch keystore Anda, Anda dapat menghilangkan bidang ini.

- aws_region (wajib) - AWS Wilayah tempat Anda membuat kampanye Amazon Personalisasi.
- ignore_failure (opsional) - Tentukan apakah plugin mengabaikan kegagalan prosesor. Untuk nilai, tentukan `true` atau`false`. Untuk lingkungan produksi Anda, kami sarankan Anda menentukan `true` untuk menghindari gangguan apa pun untuk respons kueri. Untuk lingkungan pengujian, Anda dapat menentukan `false` untuk melihat kesalahan apa pun yang dihasilkan plugin.
- external_account_iam_role_arn — Jika Anda menggunakan OpenSearch Layanan, dan sumber daya Amazon OpenSearch Personalisasi dan Layanan ada di akun yang berbeda, tentukan ARN peran yang memiliki izin untuk mengakses sumber daya Amazon Personalisasi Anda. Peran ini harus ada di akun yang sama dengan sumber daya Amazon Personalisasi Anda. Untuk informasi selengkapnya, lihat Mengkonfigurasi izin saat sumber daya berada di akun yang berbeda.

Untuk contoh kode OpenSearch Layanan, lihatMembuat pipeline di Amazon OpenSearch Service. Untuk OpenSearch contoh open source, lihatMembuat pipeline di open source OpenSearch.

Contoh metrik pipa

Saat Anda menerapkan plugin Amazon Personalize Search Ranking ke OpenSearch kueri, Anda dapat memantau plugin dengan mendapatkan metrik untuk pipeline pencarian Anda. Metrik pipeline mencakup statistik seperti jumlah permintaan gagal untuk prosesor `personalized_search_ranking` respons.

Kode berikut menunjukkan kutipan metrik pipeline yang dikembalikan dari. OpenSearch Ini hanya menunjukkan `pipelines` objek yang berisi statistik untuk dua jaringan pipa yang berbeda. Untuk setiap pipeline, Anda dapat menemukan metrik plugin Amazon Personalize Search Ranking di daftar prosesor `personalized_search_ranking` respons. Untuk contoh lengkap semua metrik, lihat Metrik saluran penelusuran.

```
{
....
....
  "pipelines": {
    "pipelineA": {
      "request": {
```

```
        "count": 0,
        "time_in_millis": 0,
        "current": 0,
        "failed": 0
      },
      "response": {
        "count": 6,
        "time_in_millis": 2246,
        "current": 0,
        "failed": 0
      },
      "request_processors": [],
      "response_processors": [
        {
          personalized_search_ranking": {
            "type": "personalized_search_ranking",
            "stats": {
              "count": <number of requests>,
              "time_in_millis": <time>,
              "current": 0,
              "failed": <number of failed requests>
            }
          }
        }
      ]
    },
    "pipelineB": {
      "request": {
        "count": 0,
        "time_in_millis": 0,
        "current": 0,
        "failed": 0
      },
      "response": {
        "count": 8,
        "time_in_millis": 2248,
        "current": 0,
        "failed": 0
      },
      "request_processors": [],
      "response_processors": [
        {
          "personalized_search_ranking": {
            "type": "personalized_search_ranking",
```

```
            "stats": {
              "count": <number of requests>,
              "time_in_millis": <time>,
              "current": 0,
              "failed": <number of failed requests>
            }
          }
        }
      ]
    }
  }
....
....
}
```

Menandai Amazon Personalisasi sumber daya

Tag adalah label yang secara opsional Anda tentukan dan kaitkan dengan AWS sumber daya, termasuk jenis sumber daya Amazon Personalize tertentu. Sumber daya dapat memiliki sebanyak 50 tag.

Tag dapat membantu Anda mengkategorikan dan mengelola sumber daya dengan berbagai cara, seperti berdasarkan tujuan, lingkungan, atau kriteria lainnya. Misalnya, Anda dapat menggunakan tag untuk membagi pendapatan antara fungsi yang berbeda, atau mengidentifikasi lingkungan pengembangan untuk sumber daya yang berbeda.

Untuk mengambil sumber daya Amazon Personalizes berdasarkan tag, Anda dapat menggunakan filter dalam GetResources pengoperasian Resource Groups Tagging API. Untuk informasi selengkapnya, lihat GetResourcesdi panduan Referensi API Penandaan API Resource Groups.

Anda dapat menambahkan tag ke jenis sumber daya Amazon Personalize berikut:

- Pekerjaan inferensi Batch
- Pekerjaan segmen Batch
- Kampanye
- Set Data
- Grup kumpulan data
- Pekerjaan impor dan ekspor dataset
- Pelacak acara
- Filter
- Rekomendasi
- Solusi
- Versi solusi

Topik

- Pedoman dan persyaratan
- Menambahkan tag ke Amazon Personalisasi sumber daya
- Menghapus tag dari Amazon Personalisasi sumber daya

- Menggunakan tag dalam kebijakan IAM

Pedoman dan persyaratan

Setiap tag terdiri dari kunci tag yang diperlukan dan nilai tag opsional, yang keduanya Anda tentukan. Kunci tag adalah label umum yang bertindak sebagai kategori untuk nilai tag yang lebih spesifik. Nilai tag bertindak sebagai deskriptor untuk kunci tag.

Misalnya, jika Anda memiliki dua versi grup kumpulan data Amazon Personalize (satu untuk pengujian internal dan satu lagi untuk produksi), Anda dapat menetapkan kunci `Environment` tag untuk kedua proyek. Nilai tag `Environment` tag mungkin `Test` untuk satu versi grup dataset dan `Production` untuk versi lainnya.

Pembatasan berikut berlaku untuk tag:

- Jumlah maksimum tag per sumber daya – 50
- Panjang kunci maksimum – 128 karakter Unicode dalam UTF-8
- Panjang nilai maksimum – 256 karakter Unicode dalam UTF-8
- Kunci dan nilai tag dapat berisi karakter berikut: A-Z, a-z, 0-9, spasi, dan _.:/= + @ — (tanda hubung). Ini adalah set standar karakter yang tersedia di seluruh AWS layanan yang mendukung tag. Beberapa layanan mendukung simbol tambahan.
- Kunci tag dan nilai tag peka huruf besar-kecil.
- Untuk setiap sumber daya terkait, setiap kunci tag harus unik dan hanya dapat memiliki satu nilai tag.
- Kunci tag dan nilai tag Anda tidak dapat memulai`aws:`. AWS layanan menerapkan tag yang dimulai dengan`aws:`, dan tag tersebut tidak dapat dimodifikasi. Mereka tidak dihitung terhadap batas tag.
- Anda tidak dapat memperbarui atau menghapus sumber daya hanya berdasarkan tagnya. Anda juga harus menentukan Nama Sumber Daya Amazon (ARN) atau ID sumber daya, tergantung pada operasi yang Anda gunakan.

Informasi tambahan

Untuk informasi selengkapnya tentang penandaan, lihat sumber daya berikut.

- AWS Prinsip Penandaan dalam Referensi AWS Umum

- AWS Strategi Penandaan (PDF yang dapat diunduh)
- AWS Kontrol Akses di Panduan Pengguna AWS IAM
- AWS Kebijakan Penandaan di AWS Organizations Panduan Pengguna

Menambahkan tag ke Amazon Personalisasi sumber daya

Anda dapat menambahkan, menampilkan, memperbarui, dan menghapus kunci dan nilai tag dari Amazon Personalisasi sumber daya dengan konsol Amazon Personalize AWS Command Line Interface ,AWS CLI(), atau. AWS SDKs Contoh berikut menunjukkan cara menambahkan tag ke grup kumpulan data Amazon Personalize. Anda dapat menambahkan tag ke sumber daya Amazon Personalize lainnya dengan cara yang sama.

Topik

- Menambahkan tag (konsol)
- Menambahkan tag (AWS CLI)
- Menambahkan tag (AWS SDKs)

Menambahkan tag (konsol)

Saat membuat sumber daya di Amazon Personalize, Anda dapat menambahkan tag opsional dengan konsol Amazon Personalize. Contoh berikut menambahkan tag ke grup dataset.

Untuk menambahkan tag ke grup dataset baru

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.
2. Pilih Buat grup kumpulan data.
3. Untuk Nama, masukkan nama.
4. Untuk Domain, pilih domain.
5. Perluas bagian Tag dan pilih Tambahkan tag baru.
6. Untuk Kunci dan Nilai, masukkan nilai yang sesuai.

   Misalnya, **`Environment`** dan**`Test`**, masing-masing.
7. Untuk menambahkan lebih banyak tag, pilih Tambahkan tag baru.

Anda dapat menambahkan hingga 50 tag ke sumber daya.

8. Pilih Berikutnya untuk terus membuat sumber daya Anda.

Menambahkan tag ke sumber daya yang ada serupa: Pilih sumber daya Anda dan gunakan bidang Tag untuk menambahkan tag Anda.

Menambahkan tag (AWS CLI)

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk menambahkan tag saat membuat sumber daya atau menambahkan tag ke sumber daya yang ada.

Topik

- Menambahkan tag saat Anda membuat sumber daya
- Untuk menambahkan tag ke sebuah sumber daya yang sudah ada

Menambahkan tag saat Anda membuat sumber daya

Untuk membuat sumber daya baru dan menambahkan tag ke dalamnya dengan AWS CLI, gunakan `create` perintah yang sesuai untuk sumber daya dan sertakan `tags` parameter dan nilai. Misalnya, perintah berikut membuat grup dataset Domain baru bernama `myDatasetGroup` untuk domain ECOMMERCE, dan menambahkan tag berikut: Kunci `Environment` tag dengan nilai `Test` tag, dan kunci `Owner` tag dan nilai. `xyzCorp`

```
aws personalize create-dataset-group \
--name myDatasetGroup \
--domain ECOMMERCE \
--tags tagKey=Environment,tagValue=Test tagKey=Owner,tagValue=xyzCorp
```

Untuk informasi tentang perintah yang dapat Anda gunakan untuk membuat sumber daya Amazon Personalisasi, lihat Referensi Perintah AWS CLI Personalisasi Amazon.

Untuk menambahkan tag ke sebuah sumber daya yang sudah ada

Untuk menambahkan tag ke sumber daya yang ada, gunakan `tag-resource` perintah. Tentukan ARN sumber daya dan berikan kunci tag dan nilai dalam parameter. `tags`

```
aws personalize tag-resource \
```

```
--resource-arn resource ARN \
--tags tagKey=key,tagValue=value
```

Menambahkan tag (AWS SDKs)

Anda dapat menggunakan AWS SDKs untuk menambahkan tag saat membuat sumber daya, atau menambahkan tag ke sumber daya yang ada.

Topik

- Menambahkan tag saat Anda membuat sumber daya
- Untuk menambahkan tag ke sebuah sumber daya yang sudah ada

Menambahkan tag saat Anda membuat sumber daya

Untuk membuat sumber daya baru dan menambahkan tag ke dalamnya dengan AWS SDKs, gunakan `create` metode yang sesuai. Gunakan `tags` parameter untuk menentukan pasangan kunci-nilai untuk setiap tag Anda. Misalnya, kode berikut membuat grup dataset Domain baru yang diberi nama `myDatasetGroup` untuk domain ECOMMERCE dan menambahkan tag berikut: Kunci `Environment` tag dengan nilai `Test` tag, dan kunci `Owner` tag dan nilai. `xyzCorp`

SDK for Python (Boto3)

```
import boto3

personalize = boto3.client('personalize')

response = personalize.create_dataset_group(
  name = 'myDatasetGroup',
  domain = 'ECOMMERCE',
  tags = [
    {
      'tagKey': 'Environment',
      'tagValue': 'Test'
    },
    {
      'tagKey': 'Owner',
      'tagValue': 'xyzCorp'
    }
  ]
)
```

```
dsg_arn = response['datasetGroupArn']

description = personalize.describe_dataset_group(datasetGroupArn = dsg_arn)
['datasetGroup']

print('Name: ' + description['name'])
print('ARN: ' + description['datasetGroupArn'])
print('Status: ' + description['status'])
```

SDK for Java 2.x

```
public static String createDomainDatasetGroup(PersonalizeClient personalizeClient,
                                               String datasetGroupName,
                                               String domain) {

    try {

        ArrayList <Tag> tags = new ArrayList<>();

        Tag tag1 = Tag.builder()
                .tagKey("Environment")
                .tagValue("Test")
                .build();
        tags.add(tag1);
        Tag tag2 = Tag.builder()
                .tagKey("Owner")
                .tagValue("xyzCorp")
                .build();
        tags.add(tag2);

        CreateDatasetGroupRequest createDatasetGroupRequest =
 CreateDatasetGroupRequest.builder()
                .name(datasetGroupName)
                .domain(domain)
                .tags(tags)
                .build();
        return
 personalizeClient.createDatasetGroup(createDatasetGroupRequest).datasetGroupArn();
    } catch (PersonalizeException e) {
        System.out.println(e.awsErrorDetails().errorMessage());
    }
    return "";
}
```

Untuk menambahkan tag ke sebuah sumber daya yang sudah ada

Kode berikut menunjukkan cara menambahkan tag ke sumber daya Amazon Personalize yang ada. Tentukan Nama Sumber Daya Amazon (ARN) sumber daya yang ingin Anda tambahkan tag dan tentukan pasangan nilai kunci untuk setiap tag Anda.

SDK for Python (Boto3)

```
import boto3
personalize = boto3.client('personalize')

add_tags_response = personalize.tag_resource(
  resourceArn = "resourceArn",
  tags = [
    {
      'tagKey': 'Environment',
      'tagValue': 'Test'
    },
    {
      'tagKey': 'Owner',
      'tagValue': 'xyzCorp'
    }
  ]
)
```

SDK for Java 2.x

```
public static void tagResource(PersonalizeClient personalizeClient,
                                                String resourceArn,
                                                String domain) {

    try {

         ArrayList <Tag> tagList = new ArrayList<>();

          Tag tag1 = Tag.builder()
                  .tagKey("Environment")
                  .tagValue("Test")
                  .build();
          tags.add(tag1);
          Tag tag2 = Tag.builder()
                  .tagKey("Owner")
                  .tagValue("xyzCorp")
```

```
                    .build();
            tags.add(tag2);

            TagResourceRequest tagResourceRequest = TagResourceRequest.builder()
                    .resourceArn(resourceArn)
                    .tags(tagList)
                    .build();

            personalizeClient.tagResource(tagResourceRequest);
            System.out.println("Tags have been added to "+ resourceArn);

        } catch (PersonalizeException e) {
            System.out.println(e.awsErrorDetails().errorMessage());
        }
        return "";
    }
```

Menghapus tag dari Amazon Personalisasi sumber daya

Anda dapat menghapus tag dari sumber daya Amazon Personalize dengan konsol Amazon Personalize atau UntagResource operasi API dengan AWS Command Line Interface () atau.AWS CLI AWS SDKs Contoh berikut menunjukkan cara menghapus tag dari grup kumpulan data Amazon Personalize. Anda dapat menghapus tag dari sumber daya Amazon Personalize lainnya dengan cara yang sama.

Topik

- Menghapus tag (konsol)
- Menghapus tag (AWS CLI)
- Menghapus tag (AWS SDKs)

Menghapus tag (konsol)

Setelah menambahkan tag ke sumber daya di Amazon Personalize, Anda dapat menghapus tag dengan konsol Amazon Personalize. Contoh berikut menghapus tag dari grup dataset

Untuk menghapus tag dari grup dataset

1. Buka konsol Amazon Personalize di https://console.aws.amazon.com/personalize/rumah dan masuk ke akun Anda.

2. Pilih grup kumpulan data Anda.
3. Di bagian bawah halaman, pilih tab Tag dan pilih Kelola tag.
4. Untuk setiap tag yang ingin Anda hapus, pilih Hapus.
5. Pilih Simpan untuk menghapus tag.

Menghapus tag (AWS CLI)

Untuk menghapus tag dari sumber daya yang ada dengan AWS CLI, gunakan `untag-resource` perintah berikut. Untuk`resource-arn`, tentukan Nama Sumber Daya Amazon (ARN) sumber daya. Untuk`tag-keys`, tentukan kunci tag yang akan dihapus.

```
aws personalize untag-resource \
--resource-arn resource ARN \
--tag-keys key1 key2
```

Menghapus tag (AWS SDKs)

Untuk menghapus tag dari sumber daya Amazon Personalize yang ada dengan AWS SDKs, gunakan operasi UntagResource API. Kode berikut menunjukkan cara menghapus beberapa tag dari grup dataset dengan SDK for Python (Boto3). Untuk`resourceArn`, tentukan Nama Sumber Daya Amazon (ARN) sumber daya. Untuk`tagKeys`, tentukan kunci tag yang akan dihapus.

```
import boto3

personalize = boto3.client('personalize')

response = personalize.untag_resource(
    resourceArn="Resource ARN",
    tagKeys=["tag1Key", "tag2Key"]
)
```

Menggunakan tag dalam kebijakan IAM

Setelah mulai menerapkan tag, Anda dapat menerapkan izin tingkat sumber daya berbasis tag ke kebijakan AWS Identity and Access Management (IAM) dan operasi API. Ini termasuk operasi yang mendukung penambahan tag ke sumber daya saat sumber daya dibuat. Dengan menggunakan tag dengan cara ini, Anda dapat menerapkan kontrol terperinci dari grup dan pengguna mana di AWS

akun Anda yang memiliki izin untuk membuat dan menandai sumber daya, dan grup dan pengguna mana yang memiliki izin untuk membuat, memperbarui, dan menghapus tag secara lebih umum.

Misalnya, Anda dapat membuat kebijakan yang memungkinkan pengguna memiliki akses penuh ke semua sumber daya Amazon Personalisasi yang namanya merupakan nilai dalam `Owner` tag untuk sumber daya.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "ModifyResourceIfOwner",
         "Effect": "Allow",
         "Action": "personalize:*",
         "Resource": "*",
         "Condition": {
            "StringEqualsIgnoreCase": {
               "aws:ResourceTag/Owner": "${aws:username}"
            }
         }
      }
   ]
}
```

Contoh berikut menunjukkan cara membuat kebijakan untuk memungkinkan pembuatan dan penghapusan kumpulan data. Operasi ini hanya diperbolehkan jika nama pengguna`johndoe`.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "personalize:CreateDataset",
                "personalize:DeleteDataset"
            ],
```

```
            "Resource": "arn:aws:personalize:*:*:dataset/*",
            "Condition": {
                "StringEquals": {"aws:username" : "johndoe"}
            }
        },
        {
            "Effect": "Allow",
            "Action": "personalize:DescribeDataset",
            "Resource": "*"
        }
    ]
}
```

Jika Anda menentukan izin tingkat sumber daya berbasis tag, izin akan segera berlaku. Ini berarti bahwa sumber daya Anda lebih aman segera setelah dibuat, dan Anda dapat dengan cepat mulai menerapkan penggunaan tag untuk sumber daya baru. Anda juga dapat menggunakan izin tingkat sumber daya untuk mengontrol kunci dan nilai tag mana yang dapat dikaitkan dengan sumber daya baru dan yang sudah ada. Untuk informasi selengkapnya, lihat Mengontrol Akses Menggunakan Tanda di AWS Panduan Pengguna IAM.

Pertanyaan yang sering diajukan untuk Amazon Personalize

Berikut ini adalah jawaban atas pertanyaan yang sering diajukan terkait dengan mengimpor data, pelatihan, penerapan model, rekomendasi, dan filter di Amazon Personalize.

Untuk pertanyaan dan jawaban selengkapnya, lihat Lembar Cheat Amazon Personalize di repositori Amazon Personalize samples.

Topik

- Impor dan manajemen data
- Membuat solusi khusus dan versi solusi
- Penerapan model (kampanye khusus)
- Rekomendasi
- Rekomendasi penyaringan

Impor dan manajemen data

Format apa yang harus dimiliki data massal saya?

Data massal Anda harus dalam format nilai yang dipisahkan koma (CSV). Baris pertama file CSV Anda harus berisi header kolom. Header kolom dalam file CSV Anda perlu dipetakan ke skema untuk membuat kumpulan data. Jika data Anda menyertakan karakter yang tidak dikodekan ASCII, file CSV Anda harus dikodekan dalam format UTF-8. Jangan lampirkan header dalam tanda kutip (“). `TIMESTAMP`dan `CREATION_TIMESTAMP` data harus dalam format waktu epoch UNIX. Untuk informasi selengkapnya tentang data stempel waktu, lihat. Data stempel waktu Untuk informasi lebih lanjut tentang skema, lihatMembuat file JSON skema untuk skema Amazon Personalize.

Untuk panduan format data lengkap, lihatMempersiapkan data pelatihan untuk Amazon Personalize. Jika Anda tidak yakin cara memformat data Anda, Anda dapat menggunakan Amazon SageMaker AI Data Wrangler (Data Wrangler) untuk menyiapkan data Anda. Untuk informasi selengkapnya, lihat Mempersiapkan dan mengimpor data massal menggunakan Amazon SageMaker AI Data Wrangler.

Berapa banyak data pelatihan yang saya butuhkan?

Untuk semua kasus penggunaan (Grup kumpulan data Domain) dan resep khusus, data interaksi Anda harus memiliki yang berikut:

- Minimal 1000 catatan interaksi item dari pengguna yang berinteraksi dengan item di katalog Anda. Interaksi ini dapat berasal dari impor massal, atau acara streaming, atau keduanya.
- Minimal 25 pengguna unik IDs dengan setidaknya dua interaksi item untuk masing-masing.

Untuk rekomendasi kualitas, kami sarankan Anda memiliki setidaknya 50.000 interaksi item dari setidaknya 1.000 pengguna dengan dua atau lebih interaksi item masing-masing.

Anda dapat memulai dengan kumpulan data interaksi Item kosong dan, ketika Anda telah merekam cukup data, buat pemberi rekomendasi Anda (Grup kumpulan data Domain) atau versi solusi khusus hanya dengan menggunakan peristiwa baru yang direkam. Beberapa resep dan kasus penggunaan mungkin memiliki persyaratan data tambahan. Untuk informasi tentang persyaratan kasus penggunaan, lihatMemilih kasus penggunaan. Untuk informasi tentang persyaratan resep, lihatMemilih resep.

Bagaimana cara memperbarui item atau atribut pengguna?

Gunakan konsol Amazon Personalize atau PutUsers operasi PutItems atau untuk mengimpor item atau pengguna dengan ID item yang sama tetapi dengan atribut yang dimodifikasi.

Bagaimana cara menghapus item atau pengguna?

Amazon Personalize tidak mendukung penghapusan item atau pengguna tertentu. Untuk memastikan bahwa item atau pengguna tidak muncul dalam rekomendasi, gunakan filter untuk mengecualikan item. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Bagaimana cara menghapus skema?

Anda dapat menghapus skema hanya dengan DeleteSchema operasi. Anda tidak dapat menggunakan konsol Amazon Personalize untuk menghapus skema.

Membuat solusi khusus dan versi solusi

Resep apa yang harus saya gunakan?

Resep Amazon Personalize yang Anda gunakan tergantung pada kasus penggunaan Anda. Untuk informasi tentang pencocokan kasus penggunaan dengan resep, lihatMemilih resep. Lembar Cheat Amazon Personalize juga menyertakan informasi kasus penggunaan dan resep.

Seberapa sering saya harus berlatih?

Kami merekomendasikan penggunaan pelatihan otomatis dengan setidaknya frekuensi pelatihan mingguan. Pelatihan otomatis memudahkan Anda untuk mempertahankan relevansi rekomendasi. Frekuensi pelatihan Anda tergantung pada kebutuhan bisnis Anda, resep yang Anda gunakan, dan seberapa sering Anda mengimpor data. Untuk informasi selengkapnya, lihat Mengkonfigurasi pelatihan otomatis. Untuk informasi tentang menjaga relevansi, lihatMempertahankan relevansi rekomendasi.

Haruskah saya menggunakan AutoML?

Tidak, kami sarankan Anda mencocokkan kasus penggunaan Anda dengan resep Personalisasi Amazon yang berbeda dan memilih resep. Untuk informasi tentang pencocokan kasus penggunaan dengan resep, lihatMemilih resep.

Penerapan model (kampanye khusus)

Apa yang harus saya tetapkan untuk minProvisionedTPS kampanye saya?

Tinggi `minProvisionedTPS` akan meningkatkan biaya Anda. Sebaiknya mulai dengan 1 untuk `minProvisionedTPS` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya.

Bagaimana cara memantau biaya kampanye saya?

Proyek Amazon Personalize Monitor menyediakan CloudWatch dasbor, metrik khusus, alarm pemanfaatan, dan fungsi pengoptimalan biaya untuk kampanye Amazon Personalize. Lihat Monitor Personalisasi Amazon di repositori Amazon Personalize samples.

Bagaimana cara mengatur throughput transaksi maksimum untuk kampanye?

Anda hanya dapat mengatur throughput minimum untuk kampanye. Saat membuat kampanye Amazon Personalize, Anda menentukan kapasitas transaksi khusus untuk membuat rekomendasi real-time bagi pengguna aplikasi Anda. Jika TPS Anda meningkat lebih dari itu`minProvisionedTPS`, Amazon Personalize secara otomatis menskalakan kapasitas yang disediakan ke atas dan ke bawah, tetapi tidak pernah di bawah. `minProvisionedTPS` Untuk informasi selengkapnya, lihat Minimum transaksi yang disediakan per detik dan auto-scaling.

Rekomendasi

Bagaimana saya bisa tahu apakah model Amazon Personalize saya menghasilkan rekomendasi kualitas?

Evaluasi kinerja versi solusi Anda dengan metrik offline dan online (lihatMengevaluasi versi solusi Amazon Personalize dengan metrik) dan pengujian online (seperti A/B testing). For more information about A/B pengujian, lihatMengukur dampak rekomendasi dengan A/B pengujian.

Bagaimana cara menghapus pekerjaan inferensi batch saya dan mengapa statusnya “aktif”?

Anda tidak dapat menghapus pekerjaan inferensi batch. Ketika status pekerjaan inferensi batch aktif, pekerjaan selesai. Anda dapat mengakses rekomendasi Anda di bucket atau folder Amazon S3 keluaran. Anda tidak akan dikenakan biaya tambahan dari pekerjaan inferensi batch setelah pekerjaan selesai. Namun Anda mungkin dikenakan biaya tambahan dari layanan lain seperti Amazon S3 untuk penyimpanan data input dan output.

Mengapa kampanye yang didukung SIM saya merekomendasikan item yang tidak serupa berdasarkan metadata?

SIMS menggunakan kumpulan data interaksi Item Anda untuk menentukan kesamaan; bukan metadata item seperti warna atau harga. SIMS mengidentifikasi kemunculan bersama item dalam riwayat pengguna di kumpulan data Interaksi Anda untuk merekomendasikan item serupa. Untuk informasi selengkapnya, lihat Resep SIMS.

Bisakah saya mendapatkan lebih dari 500 item dari satu operasi GetRecommendations API?

500 adalah jumlah maksimum item yang dapat Anda ambil dalam satuGetRecommendations. Nilai ini tidak dapat ditingkatkan.

Rekomendasi penyaringan

Mengapa rekomendasi saya tidak difilter seperti yang diharapkan?

Hal ini dapat terjadi karena berbagai alasan:

- Mungkin ada masalah dengan format atau sintaks ekspresi filter Anda. Untuk contoh ekspresi filter yang diformat dengan benar, lihatContoh ekspresi filter.
- Amazon Personalize mempertimbangkan hingga 100 interaksi terbaru per pengguna per jenis acara. Ini adalah kuota yang dapat disesuaikan. Anda dapat meminta peningkatan kuota menggunakan konsol Service Quotas. Jika Anda tidak mengimpor interaksi item untuk pengguna selama tiga bulan, filter Anda tidak lagi mempertimbangkan data historis pengguna. Untuk mempertimbangkan data ini, Anda harus mengimpor seluruh riwayat peristiwa pengguna lagi.

Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Bagaimana saya bisa menghapus barang yang sudah dibeli dari rekomendasi?

Untuk grup kumpulan data Domain ECOMMERCE, jika Anda membuat pemberi rekomendasi dengan kasus Direkomendasikan untuk Anda atau Pelanggan yang melihat X juga melihat penggunaan, Amazon Personalize secara otomatis memfilter item yang dibeli pengguna berdasarkan userID yang Anda tentukan dan peristiwa. `Purchase`

Untuk kasus penggunaan grup kumpulan data Domain lainnya atau sumber daya khusus, gunakan filter untuk menghapus item yang dibeli. Tambahkan atribut tipe `Purchased` peristiwa ke data Anda, rekam peristiwa Pembelian dengan `PutItems` operasi, dan buat filter yang menghapus item yang dibeli dari rekomendasi. Misalnya:

```
EXCLUDE ItemID WHERE Interactions.EVENT_TYPE IN ("purchased")
```

Lihat informasi yang lebih lengkap di Memfilter rekomendasi dan segmen pengguna.

Pesan kesalahan umum di Amazon Personalisasi

Bagian berikut mencantumkan dan menjelaskan beberapa pesan yang mungkin Anda temui saat menggunakan Amazon Personalize.

Topik

- Impor dan manajemen data
- Membuat solusi dan versi solusi (sumber daya khusus)
- Penerapan model (kampanye khusus)
- Rekomendasi (Grup dataset domain)
- Rekomendasi
- Rekomendasi penyaringan

Impor dan manajemen data

Pesan galat: Lokasi data tidak valid.

Pastikan Anda menggunakan sintaks yang benar untuk lokasi bucket Amazon S3 Anda. Untuk pekerjaan impor kumpulan data, gunakan sintaks berikut untuk lokasi data Anda di Amazon S3:

`s3://amzn-s3-demo-bucket/<folder path>/<CSVfilename>`

Jika file CSV Anda berada dalam folder dan Anda ingin mengunggah beberapa file dengan satu pekerjaan impor dataset, gunakan sintaks ini tanpa nama file CSV.

Pesan galat: Terjadi kesalahan (LimitExceededException) saat memanggil CreateDatasetImportJob operasi: Lebih dari 5 sumber daya dengan status PENDING atau IN_PROGRESS.

Anda dapat memiliki total 5 pekerjaan impor kumpulan data yang tertunda atau sedang berlangsung per wilayah. Kuota ini tidak dapat disesuaikan. Untuk daftar lengkap kuota untuk Amazon Personalize, lihat. Titik akhir dan kuota Amazon Personalize

Pesan galat: Gagal membuat pekerjaan impor data untuk kumpulan data... Hak istimewa yang tidak memadai untuk mengakses data di Amazon S3<dataset type>.

Berikan Amazon Personalisasi akses ke sumber daya Amazon S3 Anda dengan melampirkan kebijakan akses ke bucket Amazon S3 dan peran layanan Amazon Personalisasi Anda. Lihat Memberikan Amazon Personalisasi akses ke sumber daya Amazon S3.

Jika Anda menggunakan AWS Key Management Service (AWS KMS) untuk enkripsi, Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.

Pesan galat: Gagal membuat dataset pekerjaan impor data... Masukan CSV tidak ada kolom berikut: [COLUMN_NAME, COLUMN_NAME]<dataset type>.

Data yang Anda impor ke Amazon Personalize, termasuk nama atribut dan tipe data, harus cocok dengan skema kumpulan data tujuan. Untuk informasi selengkapnya, lihat Membuat file JSON skema untuk skema Amazon Personalize.

Pesan galat: Panjang tidak boleh lebih dari karakter untuk. <character limit><COLLUMN_NAME> Jika tidak ada nilai yang melebihi batas karakter, pastikan data Anda mengikuti pedoman pemformatan yang tercantum dalam https://docs.aws.amazon.com/personalize/ latest/dg/data -prep-formatting.html.

Periksa untuk memastikan semua nilai di kolom ini tidak melebihi batas karakter. Jika tidak ada nilai yang melebihi batas karakter, periksa bidang tekstual sebelumnya untuk hal-hal berikut:

- Pastikan setiap data tekstual dibungkus dengan tanda kutip ganda. Gunakan \ karakter untuk menghindari tanda kutip ganda atau \ karakter dalam data Anda.
- Pastikan setiap catatan dalam file CSV Anda berada pada satu baris.

Membuat solusi dan versi solusi (sumber daya khusus)

Pesan galat: Buat gagal. Dataset memiliki kurang dari 25 pengguna dengan setidaknya 2 interaksi masing-masing.

Anda harus mengimpor lebih banyak data sebelum Anda dapat melatih model. Persyaratan data minimum untuk melatih model adalah:

- Minimal 1000 catatan interaksi item dari pengguna yang berinteraksi dengan item di katalog Anda. Interaksi ini dapat berasal dari impor massal, atau acara streaming, atau keduanya.
- Minimal 25 pengguna unik IDs dengan setidaknya dua interaksi item untuk masing-masing.

Untuk rekomendasi real-time, impor lebih banyak data dengan pekerjaan impor dataset atau rekam lebih banyak peristiwa interaksi untuk pengguna Anda dengan pelacak peristiwa dan operasi.

PutEvents Untuk informasi selengkapnya tentang merekam peristiwa real-time, lihatMerekam peristiwa waktu nyata untuk memengaruhi rekomendasi.

Untuk rekomendasi batch, impor data Anda dengan pekerjaan impor dataset ketika Anda memiliki lebih banyak data. Untuk informasi selengkapnya, tentang mengimpor data massal, lihatMengimpor data pelatihan ke Amazon Personalize dataset.

Penerapan model (kampanye khusus)

Kesalahan: Tidak dapat membuat kampanye. Lebih dari 5 sumber daya dalam keadaan AKTIF. Harap hapus beberapa dan coba lagi.

Anda dapat memiliki total 5 kampanye Amazon Personalize aktif per grup kumpulan data. Kuota ini dapat disesuaikan dan Anda dapat meminta peningkatan kuota menggunakan konsol Service Quotas. Untuk daftar lengkap batasan dan kuota untuk Amazon Personalize, lihat. Titik akhir dan kuota Amazon Personalize

Rekomendasi (Grup dataset domain)

Kesalahan: Dataset memiliki kurang dari 1000 interaksi setelah memfilter berdasarkan jenis peristiwa: <event type>

Kasus penggunaan yang berbeda memerlukan jenis acara yang berbeda. Data Anda harus memiliki minimal 1000 peristiwa dengan jenis yang diperlukan untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Memilih kasus penggunaan

Rekomendasi

Pesan kesalahan pekerjaan inferensi batch: Jalur input S3 tidak valid atau Jalur keluaran S3 tidak valid

Pastikan Anda menggunakan sintaks yang benar untuk lokasi input atau output Amazon S3 Anda. Juga pastikan bahwa lokasi output Anda berbeda dari data input Anda. Itu harus berupa folder di ember Amazon S3 yang sama atau ember yang berbeda.

Gunakan sintaks berikut untuk lokasi file input di Amazon S3: **`s3://amzn-s3-demo-bucket/<folder name>/<input JSON file name>`**

Gunakan sintaks berikut untuk folder output di Amazon S3: **`s3://amzn-s3-demo-bucket/<output folder name>/`**

Rekomendasi penyaringan

Pesan galat: Tidak dapat membuat filter. Simbol masukan tidak valid: $parameterName. Placeholder tidak diperbolehkan dengan operator NOT_IN.

Anda tidak dapat menggunakan parameter placeholder dalam ekspresi filter yang menggunakan operator NOT_IN. Sebagai gantinya, gunakan operator IN dan gunakan Action yang berlawanan: gunakan Include alih-alih Exclude (atau sebaliknya).

Misalnya, jika Anda ingin menggunakan`INCLUDE ItemID WHERE Items.GENRE NOT IN ($GENRE)`, Anda dapat menggunakan `EXCLUDE ItemID WHERE Items.GENRE IN ($GENRE)` dan mendapatkan hasil yang sama.

Untuk informasi lebih lanjut tentang filter, lihat Filter elemen ekspresi.

Pesan galat: Tidak dapat membuat filter. Ekspresi Tidak Valid... saat memfilter pada bidang tipe Boolean

Anda tidak dapat membuat ekspresi filter yang memfilter menggunakan nilai dengan tipe Boolean dalam skema Anda. Untuk memfilter berdasarkan nilai Boolean, gunakan skema dengan bidang tipe `String` dan gunakan nilai `True` dan data `False` Anda. Atau Anda dapat menggunakan tipe `int` atau `long` dan nilai `0` dan`1`.

Untuk informasi lebih lanjut tentang filter, lihat Filter elemen ekspresi.

Menentukan sumber daya dengan AWS CloudFormation

Amazon Personalize terintegrasi dengan AWS CloudFormation, layanan yang membantu Anda memodelkan dan menyiapkan AWS sumber daya sehingga Anda dapat menghabiskan lebih sedikit waktu untuk membuat dan mengelola sumber daya dan infrastruktur Anda. Anda membuat templat yang menjelaskan semua AWS sumber daya yang dapat Anda tentukan (seperti grup kumpulan data Amazon Personalize). AWS CloudFormation kemudian menyediakan dan mengonfigurasi sumber daya tersebut untuk Anda.

Saat menggunakannya AWS CloudFormation, Anda dapat menggunakan kembali template untuk menyiapkan sumber daya Amazon Personalisasi secara konsisten dan berulang kali. Jelaskan sumber daya Anda sekali, lalu sediakan sumber daya yang sama berulang-ulang di beberapa Akun AWS dan Wilayah.

Topik

- Amazon Personalisasi dan templat AWS CloudFormation
- Contoh AWS CloudFormation template untuk Amazon Personalisasi sumber daya
- Pelajari lebih lanjut tentang AWS CloudFormation

Amazon Personalisasi dan templat AWS CloudFormation

Untuk menyediakan dan mengonfigurasi sumber daya untuk Amazon Personalisasi dan layanan terkait, Anda harus memahami AWS CloudFormation templat. Templat adalah file teks dengan format JSON atau YAML. Template ini menjelaskan sumber daya yang ingin Anda sediakan di AWS CloudFormation tumpukan Anda. Jika Anda tidak terbiasa dengan JSON atau YAMG, Anda dapat menggunakan AWS CloudFormation Designer untuk membantu Anda memulai dengan template. AWS CloudFormation Untuk informasi selengkapnya, lihat Apa itu AWS CloudFormation Designer? di Panduan Pengguna AWS CloudFormation .

Amazon Personalize mendukung penentuan kumpulan data, grup kumpulan data, pekerjaan impor kumpulan data, skema, dan solusi di. AWS CloudFormation Untuk informasi selengkapnya, lihat referensi jenis sumber daya Amazon Personalisasi di AWS CloudFormation Panduan Pengguna.

Contoh AWS CloudFormation template untuk Amazon Personalisasi sumber daya

Contoh AWS CloudFormation template berikut menunjukkan cara menentukan resource Amazon Personalize yang berbeda.

Topik

- CreateDatasetGroup
- CreateDataset
- CreateSchema
- CreateSolution

CreateDatasetGroup

JSON

```
{
   "AWSTemplateFormatVersion":"2010-09-09",
   "Resources":{
      "MyDatasetGroup": {
            "Type": "AWS::Personalize::DatasetGroup",
            "Properties": {
               "Name": "my-dataset-group-name"
            }
      }
   }
}
```

YAML

```
AWSTemplateFormatVersion: 2010-09-09
Resources:
  MyDatasetGroup:
    Type: 'AWS::Personalize::DatasetGroup'
    Properties:
      Name: my-dataset-group-name
```

CreateDataset

JSON

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Resources": {
    "MyDataset": {
      "Type": "AWS::Personalize::Dataset",
      "Properties": {
        "Name": "my-dataset-name",
        "DatasetType": "Interactions",
        "DatasetGroupArn": "arn:aws:personalize:us-west-2:123456789012:dataset-
group/dataset-group-name",
        "SchemaArn": "arn:aws:personalize:us-west-2:123456789012:schema/schema-
name",
        "DatasetImportJob": {
          "JobName": "my-import-job-name",
          "DataSource": {
            "DataLocation": "s3://amzn-s3-demo-bucket/file-name.csv"
          },
          "RoleArn": "arn:aws:iam::123456789012:role/personalize-role"
        }
      }
    }
  }
}
```

YAML

```
AWSTemplateFormatVersion: 2010-09-09
Resources:
  MyDataset:
    Type: 'AWS::Personalize::Dataset'
    Properties:
      Name: my-dataset-name
      DatasetType: Interactions
      DatasetGroupArn: 'arn:aws:personalize:us-west-2:123456789012:dataset-group/
dataset-group-name'
      SchemaArn: 'arn:aws:personalize:us-west-2:123456789012:schema/schema-name'
      DatasetImportJob:
        JobName: my-import-job-name
        DataSource:
```

```
          DataLocation: 's3://amzn-s3-demo-bucket/file-name.csv'
        RoleArn: 'arn:aws:iam::123456789012:role/personalize-role'
```

CreateSchema

JSON

```
{
    "AWSTemplateFormatVersion": "2010-09-09",
    "Resources": {
        "MySchema": {
            "Type": "AWS::Personalize::Schema",
            "Properties": {
                "Name": "my-schema-name",
                "Schema": "{\"type\": \"record\",\"name\": \"Interactions\",
 \"namespace\": \"com.amazonaws.personalize.schema\", \"fields\": [ { \"name\":
 \"USER_ID\", \"type\": \"string\" }, { \"name\": \"ITEM_ID\", \"type\": \"string
\" }, { \"name\": \"TIMESTAMP\", \"type\": \"long\"}], \"version\": \"1.0\"}"
            }
        }
    }
}
```

YAML

```
AWSTemplateFormatVersion: 2010-09-09
Resources:
  MySchema:
    Type: AWS::Personalize::Schema
    Properties:
      Name: "my-schema-name"
      Schema: >-
        {"type": "record","name": "Interactions", "namespace":
        "com.amazonaws.personalize.schema", "fields": [ { "name": "USER_ID",
        "type": "string" }, { "name": "ITEM_ID", "type": "string" }, { "name":
        "TIMESTAMP", "type": "long"}], "version": "1.0"}
```

CreateSolution

JSON

```
{
    "AWSTemplateFormatVersion": "2010-09-09",
    "Resources": {
        "MySolution": {
            "Type": "AWS::Personalize::Solution",
            "Properties": {
               "Name": "my-solution-name",
               "DatasetGroupArn": "arn:aws:personalize:us-
west-2:123456789012:dataset-group/my-dataset-group-name",
               "RecipeArn": "arn:aws:personalize:::recipe/aws-user-personalization",
               "SolutionConfig": {
                  "EventValueThreshold" : ".05"
                }
            }
         }
    }
}
```

YAML

```
AWSTemplateFormatVersion: 2010-09-09
Resources:
  MySolution:
    Type: 'AWS::Personalize::Solution'
    Properties:
      Name: my-solution-name
      DatasetGroupArn: >-
        arn:aws:personalize:us-west-2:123456789012:dataset-group/my-dataset-group-
name
      RecipeArn: 'arn:aws:personalize:::recipe/aws-user-personalization'
      SolutionConfig:
        EventValueThreshold: '.05'
```

Pelajari lebih lanjut tentang AWS CloudFormation

Untuk mempelajari selengkapnya AWS CloudFormation, lihat sumber daya berikut:

- AWS CloudFormation
- AWS CloudFormation panduan pengguna
- AWS CloudFormation Referensi API
- AWS CloudFormation Panduan pengguna Antarmuka Baris Perintah

Contoh kode untuk Amazon Personalisasi menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara menggunakan Amazon Personalize dengan AWS software development kit (SDK).

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Contoh kode

- Contoh kode untuk Amazon Personalisasi menggunakan AWS SDKs
  - Contoh dasar untuk Amazon Personalize menggunakan AWS SDKs
    - Tindakan untuk Amazon Personalisasi menggunakan AWS SDKs
      - Gunakan CreateBatchInferenceJob dengan AWS SDK
      - Gunakan CreateBatchSegmentJob dengan AWS SDK
      - Gunakan CreateCampaign dengan AWS SDK
      - Gunakan CreateDataset dengan AWS SDK
      - Gunakan CreateDatasetExportJob dengan AWS SDK
      - Gunakan CreateDatasetGroup dengan AWS SDK
      - Gunakan CreateDatasetImportJob dengan AWS SDK
      - Gunakan CreateEventTracker dengan AWS SDK
      - Gunakan CreateFilter dengan AWS SDK
      - Gunakan CreateRecommender dengan AWS SDK
      - Gunakan CreateSchema dengan AWS SDK
      - Gunakan CreateSolution dengan AWS SDK
      - Gunakan CreateSolutionVersion dengan AWS SDK
      - Gunakan DeleteCampaign dengan AWS SDK
      - Gunakan DeleteEventTracker dengan AWS SDK
      - Gunakan DeleteSolution dengan AWS SDK
      - Gunakan DescribeCampaign dengan AWS SDK
      - Gunakan DescribeRecipe dengan AWS SDK

        - Gunakan DescribeSolution dengan AWS SDK
        - Gunakan ListCampaigns dengan AWS SDK
        - Gunakan ListDatasetGroups dengan AWS SDK
        - Gunakan ListRecipes dengan AWS SDK
        - Gunakan ListSolutions dengan AWS SDK
        - Gunakan UpdateCampaign dengan AWS SDK
- Contoh kode untuk Amazon Personalize Events menggunakan AWS SDKs
  - Contoh dasar untuk Amazon Personalize Events menggunakan AWS SDKs
    - Tindakan untuk Amazon Personalisasi Acara menggunakan AWS SDKs
      - Gunakan PutEvents dengan AWS SDK
      - Gunakan PutItems dengan AWS SDK
      - Gunakan PutUsers dengan AWS SDK
- Contoh kode untuk Amazon Personalize Runtime menggunakan AWS SDKs
  - Contoh dasar untuk Amazon Personalize Runtime menggunakan AWS SDKs
    - Tindakan untuk Amazon Personalisasi Runtime menggunakan AWS SDKs
      - Gunakan GetPersonalizedRanking dengan AWS SDK
      - Gunakan GetRecommendations dengan AWS SDK

Contoh kode untuk Amazon Personalisasi menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara menggunakan Amazon Personalize dengan AWS software development kit (SDK).

Tindakan merupakan kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Sementara tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks dalam skenario terkait.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Contoh kode

- Contoh dasar untuk Amazon Personalize menggunakan AWS SDKs
  - Tindakan untuk Amazon Personalisasi menggunakan AWS SDKs
    - Gunakan CreateBatchInferenceJob dengan AWS SDK
    - Gunakan CreateBatchSegmentJob dengan AWS SDK
    - Gunakan CreateCampaign dengan AWS SDK
    - Gunakan CreateDataset dengan AWS SDK
    - Gunakan CreateDatasetExportJob dengan AWS SDK
    - Gunakan CreateDatasetGroup dengan AWS SDK
    - Gunakan CreateDatasetImportJob dengan AWS SDK
    - Gunakan CreateEventTracker dengan AWS SDK
    - Gunakan CreateFilter dengan AWS SDK
    - Gunakan CreateRecommender dengan AWS SDK
    - Gunakan CreateSchema dengan AWS SDK
    - Gunakan CreateSolution dengan AWS SDK
    - Gunakan CreateSolutionVersion dengan AWS SDK
    - Gunakan DeleteCampaign dengan AWS SDK
    - Gunakan DeleteEventTracker dengan AWS SDK
    - Gunakan DeleteSolution dengan AWS SDK
    - Gunakan DescribeCampaign dengan AWS SDK
    - Gunakan DescribeRecipe dengan AWS SDK
    - Gunakan DescribeSolution dengan AWS SDK
    - Gunakan ListCampaigns dengan AWS SDK
    - Gunakan ListDatasetGroups dengan AWS SDK
    - Gunakan ListRecipes dengan AWS SDK
    - Gunakan ListSolutions dengan AWS SDK
    - Gunakan UpdateCampaign dengan AWS SDK

Contoh dasar untuk Amazon Personalize menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara menggunakan dasar-dasar Amazon Personalize with. AWS SDKs

Contoh

- Tindakan untuk Amazon Personalisasi menggunakan AWS SDKs
  - Gunakan CreateBatchInferenceJob dengan AWS SDK
  - Gunakan CreateBatchSegmentJob dengan AWS SDK
  - Gunakan CreateCampaign dengan AWS SDK
  - Gunakan CreateDataset dengan AWS SDK
  - Gunakan CreateDatasetExportJob dengan AWS SDK
  - Gunakan CreateDatasetGroup dengan AWS SDK
  - Gunakan CreateDatasetImportJob dengan AWS SDK
  - Gunakan CreateEventTracker dengan AWS SDK
  - Gunakan CreateFilter dengan AWS SDK
  - Gunakan CreateRecommender dengan AWS SDK
  - Gunakan CreateSchema dengan AWS SDK
  - Gunakan CreateSolution dengan AWS SDK
  - Gunakan CreateSolutionVersion dengan AWS SDK
  - Gunakan DeleteCampaign dengan AWS SDK
  - Gunakan DeleteEventTracker dengan AWS SDK
  - Gunakan DeleteSolution dengan AWS SDK
  - Gunakan DescribeCampaign dengan AWS SDK
  - Gunakan DescribeRecipe dengan AWS SDK
  - Gunakan DescribeSolution dengan AWS SDK
  - Gunakan ListCampaigns dengan AWS SDK
  - Gunakan ListDatasetGroups dengan AWS SDK
  - Gunakan ListRecipes dengan AWS SDK
  - Gunakan ListSolutions dengan AWS SDK
  - Gunakan UpdateCampaign dengan AWS SDK

Tindakan untuk Amazon Personalisasi menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara melakukan tindakan Amazon Personalize individual dengan. AWS SDKs Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan instruksi untuk mengatur dan menjalankan kode.

Contoh berikut hanya mencakup tindakan yang paling umum digunakan. Untuk daftar lengkapnya, lihat Referensi API Personalisasi Amazon.

Contoh

- Gunakan CreateBatchInferenceJob dengan AWS SDK
- Gunakan CreateBatchSegmentJob dengan AWS SDK
- Gunakan CreateCampaign dengan AWS SDK
- Gunakan CreateDataset dengan AWS SDK
- Gunakan CreateDatasetExportJob dengan AWS SDK
- Gunakan CreateDatasetGroup dengan AWS SDK
- Gunakan CreateDatasetImportJob dengan AWS SDK
- Gunakan CreateEventTracker dengan AWS SDK
- Gunakan CreateFilter dengan AWS SDK
- Gunakan CreateRecommender dengan AWS SDK
- Gunakan CreateSchema dengan AWS SDK
- Gunakan CreateSolution dengan AWS SDK
- Gunakan CreateSolutionVersion dengan AWS SDK
- Gunakan DeleteCampaign dengan AWS SDK
- Gunakan DeleteEventTracker dengan AWS SDK
- Gunakan DeleteSolution dengan AWS SDK
- Gunakan DescribeCampaign dengan AWS SDK
- Gunakan DescribeRecipe dengan AWS SDK
- Gunakan DescribeSolution dengan AWS SDK
- Gunakan ListCampaigns dengan AWS SDK
- Gunakan ListDatasetGroups dengan AWS SDK
- Gunakan ListRecipes dengan AWS SDK

- Gunakan ListSolutions dengan AWS SDK
- Gunakan UpdateCampaign dengan AWS SDK

Gunakan **`CreateBatchInferenceJob`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateBatchInferenceJob`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
        public static String createPersonalizeBatchInferenceJob(PersonalizeClient
personalizeClient,
                        String solutionVersionArn,
                        String jobName,
                        String s3InputDataSourcePath,
                        String s3DataDestinationPath,
                        String roleArn,
                        String explorationWeight,
                        String explorationItemAgeCutOff) {

                long waitInMilliseconds = 60 * 1000;
                String status;
                String batchInferenceJobArn;

                try {

                        // Set up data input and output parameters.
                        S3DataConfig inputSource = S3DataConfig.builder()
                                        .path(s3InputDataSourcePath)
                                        .build();

                        S3DataConfig outputDestination = S3DataConfig.builder()
                                        .path(s3DataDestinationPath)
                                        .build();
```

```
                        BatchInferenceJobInput jobInput =
BatchInferenceJobInput.builder()
                                        .s3DataSource(inputSource)
                                        .build();

                        BatchInferenceJobOutput jobOutputLocation =
BatchInferenceJobOutput.builder()
                                        .s3DataDestination(outputDestination)
                                        .build();

                        // Optional code to build the User-Personalization
specific item exploration
                        // config.
                        HashMap<String, String> explorationConfig = new
HashMap<>();

                        explorationConfig.put("explorationWeight",
explorationWeight);
                        explorationConfig.put("explorationItemAgeCutOff",
explorationItemAgeCutOff);

                        BatchInferenceJobConfig jobConfig =
BatchInferenceJobConfig.builder()
                                        .itemExplorationConfig(explorationConfig)
                                        .build();

                        // End optional User-Personalization recipe specific
code.

                        CreateBatchInferenceJobRequest
createBatchInferenceJobRequest = CreateBatchInferenceJobRequest
                                        .builder()
                                        .solutionVersionArn(solutionVersionArn)
                                        .jobInput(jobInput)
                                        .jobOutput(jobOutputLocation)
                                        .jobName(jobName)
                                        .roleArn(roleArn)
                                        .batchInferenceJobConfig(jobConfig) //
Optional
                                        .build();

                        batchInferenceJobArn =
personalizeClient.createBatchInferenceJob(createBatchInferenceJobRequest)
```

```
                                        .batchInferenceJobArn();

                        DescribeBatchInferenceJobRequest
describeBatchInferenceJobRequest = DescribeBatchInferenceJobRequest
                                        .builder()

.batchInferenceJobArn(batchInferenceJobArn)
                                        .build();

                        long maxTime = Instant.now().getEpochSecond() + 3 * 60 *
60;
                        while (Instant.now().getEpochSecond() < maxTime) {

                                BatchInferenceJob batchInferenceJob =
personalizeClient

.describeBatchInferenceJob(describeBatchInferenceJobRequest)
                                                .batchInferenceJob();

                                status = batchInferenceJob.status();
                                System.out.println("Batch inference job status: "
+ status);

                                if (status.equals("ACTIVE") ||
status.equals("CREATE FAILED")) {
                                        break;
                                }
                                try {
                                        Thread.sleep(waitInMilliseconds);
                                } catch (InterruptedException e) {
                                        System.out.println(e.getMessage());
                                }
                        }
                        return batchInferenceJobArn;

                } catch (PersonalizeException e) {
                        System.out.println(e.awsErrorDetails().errorMessage());
                }
                return "";
        }
```

- Untuk detail API, lihat CreateBatchInferenceJobdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateBatchInferenceJobCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the batch inference job's parameters.

export const createBatchInferenceJobParam = {
  jobName: "JOB_NAME",
  jobInput: {
    s3DataSource: {
      path: "INPUT_PATH",
    },
  },
  jobOutput: {
    s3DataDestination: {
      path: "OUTPUT_PATH",
    },
  },
  roleArn: "ROLE_ARN",
  solutionVersionArn: "SOLUTION_VERSION_ARN",
  numResults: 20,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateBatchInferenceJobCommand(createBatchInferenceJobParam),
    );
    console.log("Success", response);
```

```
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateBatchInferenceJobdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateBatchSegmentJob`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateBatchSegmentJob`.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateBatchSegmentJobCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the batch segment job's parameters.

export const createBatchSegmentJobParam = {
  jobName: "NAME",
```

```
  jobInput: {
    s3DataSource: {
      path: "INPUT_PATH",
    },
  },
  jobOutput: {
    s3DataDestination: {
      path: "OUTPUT_PATH",
    },
  },
  roleArn: "ROLE_ARN",
  solutionVersionArn: "SOLUTION_VERSION_ARN",
  numResults: 20,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateBatchSegmentJobCommand(createBatchSegmentJobParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateBatchSegmentJobdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateCampaign`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateCampaign`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static String createPersonalCompaign(PersonalizeClient
personalizeClient, String solutionVersionArn,
           String name) {

       try {
           CreateCampaignRequest createCampaignRequest =
CreateCampaignRequest.builder()
                   .minProvisionedTPS(1)
                   .solutionVersionArn(solutionVersionArn)
                   .name(name)
                   .build();

           CreateCampaignResponse campaignResponse =
personalizeClient.createCampaign(createCampaignRequest);
           System.out.println("The campaign ARN is " +
campaignResponse.campaignArn());
           return campaignResponse.campaignArn();
       } catch (PersonalizeException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
           System.exit(1);
       }
       return null;
   }
```

- Untuk detail API, lihat CreateCampaigndi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.

import { CreateCampaignCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the campaign's parameters.
export const createCampaignParam = {
  solutionVersionArn: "SOLUTION_VERSION_ARN" /* required */,
  name: "NAME" /* required */,
  minProvisionedTPS: 1 /* optional integer */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateCampaignCommand(createCampaignParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateCampaigndi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateDataset`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateDataset`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
public static String createDataset(PersonalizeClient personalizeClient,
        String datasetName,
        String datasetGroupArn,
        String datasetType,
        String schemaArn) {
    try {
        CreateDatasetRequest request = CreateDatasetRequest.builder()
                .name(datasetName)
                .datasetGroupArn(datasetGroupArn)
                .datasetType(datasetType)
                .schemaArn(schemaArn)
                .build();

        String datasetArn = personalizeClient.createDataset(request)
                .datasetArn();
        System.out.println("Dataset " + datasetName + " created.");
        return datasetArn;

    } catch (PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return "";
}
```

- Untuk detail API, lihat CreateDatasetdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateDatasetCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the dataset's parameters.
export const createDatasetParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  datasetType: "DATASET_TYPE" /* required */,
  name: "NAME" /* required */,
  schemaArn: "SCHEMA_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetCommand(createDatasetParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateDatasetdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateDatasetExportJob`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateDatasetExportJob`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static String createDatasetExportJob(PersonalizeClient
personalizeClient,
           String jobName,
           String datasetArn,
           IngestionMode ingestionMode,
           String roleArn,
           String s3BucketPath,
           String kmsKeyArn) {

       long waitInMilliseconds = 30 * 1000; // 30 seconds
       String status = null;

       try {

           S3DataConfig exportS3DataConfig =
S3DataConfig.builder().path(s3BucketPath).kmsKeyArn(kmsKeyArn).build();
           DatasetExportJobOutput jobOutput =
DatasetExportJobOutput.builder().s3DataDestination(exportS3DataConfig)
                   .build();
```

```
            CreateDatasetExportJobRequest createRequest =
CreateDatasetExportJobRequest.builder()
                    .jobName(jobName)
                    .datasetArn(datasetArn)
                    .ingestionMode(ingestionMode)
                    .jobOutput(jobOutput)
                    .roleArn(roleArn)
                    .build();

            String datasetExportJobArn =
personalizeClient.createDatasetExportJob(createRequest).datasetExportJobArn();

            DescribeDatasetExportJobRequest describeDatasetExportJobRequest =
DescribeDatasetExportJobRequest.builder()
                    .datasetExportJobArn(datasetExportJobArn)
                    .build();

            long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            while (Instant.now().getEpochSecond() < maxTime) {

                DatasetExportJob datasetExportJob = personalizeClient

.describeDatasetExportJob(describeDatasetExportJobRequest)
                        .datasetExportJob();

                status = datasetExportJob.status();
                System.out.println("Export job status: " + status);

                if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
                    return status;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }
        } catch (PersonalizeException e) {
            System.out.println(e.awsErrorDetails().errorMessage());
        }
        return "";
    }
```

- Untuk detail API, lihat CreateDatasetExportJobdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateDatasetExportJobCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the export job parameters.
export const datasetExportJobParam = {
  datasetArn: "DATASET_ARN" /* required */,
  jobOutput: {
    s3DataDestination: {
      path: "S3_DESTINATION_PATH" /* required */,
      //kmsKeyArn: 'ARN'  /* include if your bucket uses AWS KMS for encryption
    },
  },
  jobName: "NAME" /* required */,
  roleArn: "ROLE_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetExportJobCommand(datasetExportJobParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
```

```
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateDatasetExportJobdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateDatasetGroup`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateDatasetGroup`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static String createDatasetGroup(PersonalizeClient personalizeClient,
String datasetGroupName) {

       try {
           CreateDatasetGroupRequest createDatasetGroupRequest =
CreateDatasetGroupRequest.builder()
                   .name(datasetGroupName)
                   .build();
           return
personalizeClient.createDatasetGroup(createDatasetGroupRequest).datasetGroupArn();
       } catch (PersonalizeException e) {
           System.out.println(e.awsErrorDetails().errorMessage());
       }
       return "";
   }
```

Buat grup dataset domain.

```
   public static String createDomainDatasetGroup(PersonalizeClient
personalizeClient,
           String datasetGroupName,
           String domain) {

       try {
           CreateDatasetGroupRequest createDatasetGroupRequest =
CreateDatasetGroupRequest.builder()
                   .name(datasetGroupName)
                   .domain(domain)
                   .build();
           return
personalizeClient.createDatasetGroup(createDatasetGroupRequest).datasetGroupArn();
       } catch (PersonalizeException e) {
           System.out.println(e.awsErrorDetails().errorMessage());
       }
       return "";
   }
```

- Untuk detail API, lihat CreateDatasetGroupdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.

import { CreateDatasetGroupCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";
```

```
// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the dataset group parameters.
export const createDatasetGroupParam = {
  name: "NAME" /* required */,
};

export const run = async (createDatasetGroupParam) => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetGroupCommand(createDatasetGroupParam),
    );
    console.log("Success", response);
    return "Run successfully"; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run(createDatasetGroupParam);
```

Buat grup dataset domain.

```
// Get service clients module and commands using ES6 syntax.
import { CreateDatasetGroupCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the domain dataset group parameters.
export const domainDatasetGroupParams = {
  name: "NAME" /* required */,
  domain:
    "DOMAIN" /* required for a domain dsg, specify ECOMMERCE or VIDEO_ON_DEMAND
 */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetGroupCommand(domainDatasetGroupParams),
```

```
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateDatasetGroupdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateDatasetImportJob`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateDatasetImportJob`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static String createPersonalizeDatasetImportJob(PersonalizeClient
personalizeClient,
           String jobName,
           String datasetArn,
           String s3BucketPath,
           String roleArn) {

       long waitInMilliseconds = 60 * 1000;
       String status;
       String datasetImportJobArn;
```

```
        try {
            DataSource importDataSource = DataSource.builder()
                    .dataLocation(s3BucketPath)
                    .build();

            CreateDatasetImportJobRequest createDatasetImportJobRequest =
CreateDatasetImportJobRequest.builder()
                    .datasetArn(datasetArn)
                    .dataSource(importDataSource)
                    .jobName(jobName)
                    .roleArn(roleArn)
                    .build();

            datasetImportJobArn =
personalizeClient.createDatasetImportJob(createDatasetImportJobRequest)
                    .datasetImportJobArn();
            DescribeDatasetImportJobRequest describeDatasetImportJobRequest =
DescribeDatasetImportJobRequest.builder()
                    .datasetImportJobArn(datasetImportJobArn)
                    .build();

            long maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            while (Instant.now().getEpochSecond() < maxTime) {

                DatasetImportJob datasetImportJob = personalizeClient

.describeDatasetImportJob(describeDatasetImportJobRequest)
                        .datasetImportJob();

                status = datasetImportJob.status();
                System.out.println("Dataset import job status: " + status);

                if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }
            return datasetImportJobArn;
```

```
        } catch (PersonalizeException e) {
            System.out.println(e.awsErrorDetails().errorMessage());
        }
        return "";
    }
```

- Untuk detail API, lihat CreateDatasetImportJobdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateDatasetImportJobCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the dataset import job parameters.
export const datasetImportJobParam = {
  datasetArn: "DATASET_ARN" /* required */,
  dataSource: {
    /* required */
    dataLocation: "S3_PATH",
  },
  jobName: "NAME" /* required */,
  roleArn: "ROLE_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateDatasetImportJobCommand(datasetImportJobParam),
    );
```

```
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateDatasetImportJobdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateEventTracker`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateEventTracker`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static String createEventTracker(PersonalizeClient personalizeClient,
String eventTrackerName,
           String datasetGroupArn) {

       String eventTrackerId = "";
       String eventTrackerArn;
       long maxTime = 3 * 60 * 60; // 3 hours
       long waitInMilliseconds = 20 * 1000; // 20 seconds
       String status;

       try {
```

```
            CreateEventTrackerRequest createEventTrackerRequest =
CreateEventTrackerRequest.builder()
                    .name(eventTrackerName)
                    .datasetGroupArn(datasetGroupArn)
                    .build();

            CreateEventTrackerResponse createEventTrackerResponse =
personalizeClient
                    .createEventTracker(createEventTrackerRequest);

            eventTrackerArn = createEventTrackerResponse.eventTrackerArn();
            eventTrackerId = createEventTrackerResponse.trackingId();
            System.out.println("Event tracker ARN: " + eventTrackerArn);
            System.out.println("Event tracker ID: " + eventTrackerId);

            maxTime = Instant.now().getEpochSecond() + maxTime;

            DescribeEventTrackerRequest describeRequest =
DescribeEventTrackerRequest.builder()
                    .eventTrackerArn(eventTrackerArn)
                    .build();

            while (Instant.now().getEpochSecond() < maxTime) {

                status =
personalizeClient.describeEventTracker(describeRequest).eventTracker().status();
                System.out.println("EventTracker status: " + status);

                if (status.equals("ACTIVE") || status.equals("CREATE FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }
            return eventTrackerId;
        } catch (PersonalizeException e) {
            System.out.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return eventTrackerId;
    }
```

- Untuk detail API, lihat CreateEventTrackerdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateEventTrackerCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the event tracker's parameters.
export const createEventTrackerParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  name: "NAME" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateEventTrackerCommand(createEventTrackerParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateEventTrackerdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateFilter`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateFilter`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
public static String createFilter(PersonalizeClient personalizeClient,
        String filterName,
        String datasetGroupArn,
        String filterExpression) {
    try {
        CreateFilterRequest request = CreateFilterRequest.builder()
                .name(filterName)
                .datasetGroupArn(datasetGroupArn)
                .filterExpression(filterExpression)
                .build();

        return personalizeClient.createFilter(request).filterArn();
    } catch (PersonalizeException e) {
        System.err.println(e.awsErrorDetails().errorMessage());
        System.exit(1);
    }
    return "";
}
```

- Untuk detail API, lihat CreateFilterdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateFilterCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the filter's parameters.
export const createFilterParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  name: "NAME" /* required */,
  filterExpression: "FILTER_EXPRESSION" /*required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateFilterCommand(createFilterParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateFilterdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateRecommender`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateRecommender`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static String createRecommender(PersonalizeClient personalizeClient,
            String name,
            String datasetGroupArn,
            String recipeArn) {

        long maxTime = 0;
        long waitInMilliseconds = 30 * 1000; // 30 seconds
        String recommenderStatus = "";

        try {
            CreateRecommenderRequest createRecommenderRequest =
CreateRecommenderRequest.builder()
                    .datasetGroupArn(datasetGroupArn)
                    .name(name)
                    .recipeArn(recipeArn)
                    .build();

            CreateRecommenderResponse recommenderResponse = personalizeClient
                    .createRecommender(createRecommenderRequest);
            String recommenderArn = recommenderResponse.recommenderArn();
            System.out.println("The recommender ARN is " + recommenderArn);

            DescribeRecommenderRequest describeRecommenderRequest =
DescribeRecommenderRequest.builder()
```

```
                    .recommenderArn(recommenderArn)
                    .build();

            maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            while (Instant.now().getEpochSecond() < maxTime) {

                recommenderStatus =
personalizeClient.describeRecommender(describeRecommenderRequest).recommender()
                        .status();
                System.out.println("Recommender status: " + recommenderStatus);

                if (recommenderStatus.equals("ACTIVE") ||
recommenderStatus.equals("CREATE FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }
            return recommenderArn;

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

- Untuk detail API, lihat CreateRecommenderdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateRecommenderCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the recommender's parameters.
export const createRecommenderParam = {
  name: "NAME" /* required */,
  recipeArn: "RECIPE_ARN" /* required */,
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateRecommenderCommand(createRecommenderParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateRecommenderdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateSchema`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateSchema`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static String createSchema(PersonalizeClient personalizeClient, String
schemaName, String filePath) {

       String schema = null;
       try {
           schema = new String(Files.readAllBytes(Paths.get(filePath)));
       } catch (IOException e) {
           System.out.println(e.getMessage());
       }

       try {
           CreateSchemaRequest createSchemaRequest =
CreateSchemaRequest.builder()
                   .name(schemaName)
                   .schema(schema)
                   .build();

           String schemaArn =
personalizeClient.createSchema(createSchemaRequest).schemaArn();

           System.out.println("Schema arn: " + schemaArn);

           return schemaArn;

       } catch (PersonalizeException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
           System.exit(1);
       }
       return "";
   }
```

Buat skema dengan domain.

```
   public static String createDomainSchema(PersonalizeClient personalizeClient,
String schemaName, String domain,
           String filePath) {

       String schema = null;
       try {
           schema = new String(Files.readAllBytes(Paths.get(filePath)));
       } catch (IOException e) {
           System.out.println(e.getMessage());
       }

       try {
           CreateSchemaRequest createSchemaRequest =
CreateSchemaRequest.builder()
                   .name(schemaName)
                   .domain(domain)
                   .schema(schema)
                   .build();

           String schemaArn =
personalizeClient.createSchema(createSchemaRequest).schemaArn();

           System.out.println("Schema arn: " + schemaArn);

           return schemaArn;

       } catch (PersonalizeException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
           System.exit(1);
       }
       return "";
   }
```

- Untuk detail API, lihat CreateSchemadi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateSchemaCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

import fs from "node:fs";

const schemaFilePath = "SCHEMA_PATH";
let mySchema = "";

try {
  mySchema = fs.readFileSync(schemaFilePath).toString();
} catch (err) {
  mySchema = "TEST"; // For unit tests.
}
// Set the schema parameters.
export const createSchemaParam = {
  name: "NAME" /* required */,
  schema: mySchema /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSchemaCommand(createSchemaParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
```

```
  }
};
run();
```

Buat skema dengan domain.

```
// Get service clients module and commands using ES6 syntax.
import { CreateSchemaCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";

// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

import fs from "node:fs";

const schemaFilePath = "SCHEMA_PATH";
let mySchema = "";

try {
  mySchema = fs.readFileSync(schemaFilePath).toString();
} catch (err) {
  mySchema = "TEST"; // for unit tests.
}

// Set the domain schema parameters.
export const createDomainSchemaParam = {
  name: "NAME" /* required */,
  schema: mySchema /* required */,
  domain:
    "DOMAIN" /* required for a domain dataset group, specify ECOMMERCE or
 VIDEO_ON_DEMAND */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSchemaCommand(createDomainSchemaParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
```

```
  }
};
run();
```

- Untuk detail API, lihat CreateSchemadi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateSolution`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateSolution`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static String createPersonalizeSolution(PersonalizeClient
personalizeClient,
           String datasetGroupArn,
           String solutionName,
           String recipeArn) {

       try {
           CreateSolutionRequest solutionRequest =
CreateSolutionRequest.builder()
                   .name(solutionName)
                   .datasetGroupArn(datasetGroupArn)
                   .recipeArn(recipeArn)
                   .build();

           CreateSolutionResponse solutionResponse =
personalizeClient.createSolution(solutionRequest);
```

```
            return solutionResponse.solutionArn();

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

- Untuk detail API, lihat CreateSolutiondi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateSolutionCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the solution parameters.
export const createSolutionParam = {
  datasetGroupArn: "DATASET_GROUP_ARN" /* required */,
  recipeArn: "RECIPE_ARN" /* required */,
  name: "NAME" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSolutionCommand(createSolutionParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
```

```
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateSolutiondi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`CreateSolutionVersion`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`CreateSolutionVersion`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static String createPersonalizeSolutionVersion(PersonalizeClient
personalizeClient, String solutionArn) {
       long maxTime = 0;
       long waitInMilliseconds = 30 * 1000; // 30 seconds
       String solutionStatus = "";
       String solutionVersionStatus = "";
       String solutionVersionArn = "";

       try {
           DescribeSolutionRequest describeSolutionRequest =
DescribeSolutionRequest.builder()
                   .solutionArn(solutionArn)
                   .build();
```

```
            maxTime = Instant.now().getEpochSecond() + 3 * 60 * 60;

            // Wait until solution is active.
            while (Instant.now().getEpochSecond() < maxTime) {

                solutionStatus =
personalizeClient.describeSolution(describeSolutionRequest).solution().status();
                System.out.println("Solution status: " + solutionStatus);

                if (solutionStatus.equals("ACTIVE") ||
solutionStatus.equals("CREATE FAILED")) {
                    break;
                }
                try {
                    Thread.sleep(waitInMilliseconds);
                } catch (InterruptedException e) {
                    System.out.println(e.getMessage());
                }
            }

            if (solutionStatus.equals("ACTIVE")) {

                CreateSolutionVersionRequest createSolutionVersionRequest =
CreateSolutionVersionRequest.builder()
                        .solutionArn(solutionArn)
                        .build();

                CreateSolutionVersionResponse createSolutionVersionResponse =
personalizeClient
                        .createSolutionVersion(createSolutionVersionRequest);
                solutionVersionArn =
createSolutionVersionResponse.solutionVersionArn();

                System.out.println("Solution version ARN: " +
solutionVersionArn);

                DescribeSolutionVersionRequest describeSolutionVersionRequest =
DescribeSolutionVersionRequest.builder()
                        .solutionVersionArn(solutionVersionArn)
                        .build();

                while (Instant.now().getEpochSecond() < maxTime) {
```

```
                    solutionVersionStatus =
personalizeClient.describeSolutionVersion(describeSolutionVersionRequest)
                            .solutionVersion().status();
                    System.out.println("Solution version status: " +
solutionVersionStatus);

                    if (solutionVersionStatus.equals("ACTIVE") ||
solutionVersionStatus.equals("CREATE FAILED")) {
                        break;
                    }
                    try {
                        Thread.sleep(waitInMilliseconds);
                    } catch (InterruptedException e) {
                        System.out.println(e.getMessage());
                    }
                }
                return solutionVersionArn;
            }
        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

- Untuk detail API, lihat CreateSolutionVersiondi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { CreateSolutionVersionCommand } from "@aws-sdk/client-personalize";
import { personalizeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
```

```
// const personalizeClient = new PersonalizeClient({ region: "REGION"});

// Set the solution version parameters.
export const solutionVersionParam = {
  solutionArn: "SOLUTION_ARN" /* required */,
};

export const run = async () => {
  try {
    const response = await personalizeClient.send(
      new CreateSolutionVersionCommand(solutionVersionParam),
    );
    console.log("Success", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat CreateSolutionVersiondi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`DeleteCampaign`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`DeleteCampaign`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static void deleteSpecificCampaign(PersonalizeClient
 personalizeClient, String campaignArn) {
        try {
            DeleteCampaignRequest campaignRequest =
 DeleteCampaignRequest.builder()
                    .campaignArn(campaignArn)
                    .build();

            personalizeClient.deleteCampaign(campaignRequest);
            System.out.println("Delete request sent successfully.");
        } catch (PersonalizeException e) {
            System.err.println("Error deleting campaign: " +
 e.awsErrorDetails().errorMessage());
            throw new RuntimeException(e);
        }
    }
}
```

- Untuk detail API, lihat DeleteCampaigndi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`DeleteEventTracker`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`DeleteEventTracker`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static void deleteEventTracker(PersonalizeClient personalizeClient,
 String eventTrackerArn) {
```

```
        try {
            DeleteEventTrackerRequest deleteEventTrackerRequest =
DeleteEventTrackerRequest.builder()
                    .eventTrackerArn(eventTrackerArn)
                    .build();

            int status =
personalizeClient.deleteEventTracker(deleteEventTrackerRequest).sdkHttpResponse().status

            System.out.println("Status code:" + status);

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Untuk detail API, lihat DeleteEventTrackerdi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`DeleteSolution`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`DeleteSolution`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static void deleteGivenSolution(PersonalizeClient personalizeClient,
String solutionArn) {
```

```
        try {
            DeleteSolutionRequest solutionRequest =
DeleteSolutionRequest.builder()
                    .solutionArn(solutionArn)
                    .build();

            personalizeClient.deleteSolution(solutionRequest);
            System.out.println("Done");

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Untuk detail API, lihat DeleteSolutiondi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`DescribeCampaign`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`DescribeCampaign`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static void describeSpecificCampaign(PersonalizeClient
personalizeClient, String campaignArn) {

        try {
```

```
            DescribeCampaignRequest campaignRequest =
DescribeCampaignRequest.builder()
                    .campaignArn(campaignArn)
                    .build();

            DescribeCampaignResponse campaignResponse =
personalizeClient.describeCampaign(campaignRequest);
            Campaign myCampaign = campaignResponse.campaign();
            System.out.println("The Campaign name is " + myCampaign.name());
            System.out.println("The Campaign status is " + myCampaign.status());

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Untuk detail API, lihat DescribeCampaigndi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`DescribeRecipe`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`DescribeRecipe`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static void describeSpecificRecipe(PersonalizeClient
personalizeClient, String recipeArn) {
```

```
        try {
            DescribeRecipeRequest recipeRequest = DescribeRecipeRequest.builder()
                    .recipeArn(recipeArn)
                    .build();

            DescribeRecipeResponse recipeResponse =
personalizeClient.describeRecipe(recipeRequest);
            System.out.println("The recipe name is " +
recipeResponse.recipe().name());

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Untuk detail API, lihat DescribeRecipedi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`DescribeSolution`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`DescribeSolution`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static void describeSpecificSolution(PersonalizeClient
personalizeClient, String solutionArn) {

        try {
```

```
            DescribeSolutionRequest solutionRequest =
DescribeSolutionRequest.builder()
                    .solutionArn(solutionArn)
                    .build();

            DescribeSolutionResponse response =
personalizeClient.describeSolution(solutionRequest);
            System.out.println("The Solution name is " +
response.solution().name());

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Untuk detail API, lihat DescribeSolutiondi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`ListCampaigns`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`ListCampaigns`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static void listAllCampaigns(PersonalizeClient personalizeClient,
String solutionArn) {

        try {
```

```
            ListCampaignsRequest campaignsRequest =
ListCampaignsRequest.builder()
                    .maxResults(10)
                    .solutionArn(solutionArn)
                    .build();

            ListCampaignsResponse response =
personalizeClient.listCampaigns(campaignsRequest);
            List<CampaignSummary> campaigns = response.campaigns();
            for (CampaignSummary campaign : campaigns) {
                System.out.println("Campaign name is : " + campaign.name());
                System.out.println("Campaign ARN is : " +
campaign.campaignArn());
            }

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Untuk detail API, lihat ListCampaignsdi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`ListDatasetGroups`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`ListDatasetGroups`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static void listDSGroups(PersonalizeClient personalizeClient) {

       try {
           ListDatasetGroupsRequest groupsRequest =
ListDatasetGroupsRequest.builder()
                   .maxResults(15)
                   .build();

           ListDatasetGroupsResponse groupsResponse =
personalizeClient.listDatasetGroups(groupsRequest);
           List<DatasetGroupSummary> groups = groupsResponse.datasetGroups();
           for (DatasetGroupSummary group : groups) {
               System.out.println("The DataSet name is : " + group.name());
               System.out.println("The DataSet ARN is : " +
group.datasetGroupArn());
           }

       } catch (PersonalizeException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
           System.exit(1);
       }
   }
```

- Untuk detail API, lihat ListDatasetGroupsdi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`ListRecipes`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`ListRecipes`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static void listAllRecipes(PersonalizeClient personalizeClient) {

        try {
            ListRecipesRequest recipesRequest = ListRecipesRequest.builder()
                    .maxResults(15)
                    .build();

            ListRecipesResponse response =
personalizeClient.listRecipes(recipesRequest);
            List<RecipeSummary> recipes = response.recipes();
            for (RecipeSummary recipe : recipes) {
                System.out.println("The recipe ARN is: " + recipe.recipeArn());
                System.out.println("The recipe name is: " + recipe.name());
            }

        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Untuk detail API, lihat ListRecipesdi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`ListSolutions`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`ListSolutions`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static void listAllSolutions(PersonalizeClient personalizeClient,
String datasetGroupArn) {

       try {
           ListSolutionsRequest solutionsRequest =
ListSolutionsRequest.builder()
                   .maxResults(10)
                   .datasetGroupArn(datasetGroupArn)
                   .build();

           ListSolutionsResponse response =
personalizeClient.listSolutions(solutionsRequest);
           List<SolutionSummary> solutions = response.solutions();
           for (SolutionSummary solution : solutions) {
               System.out.println("The solution ARN is: " +
solution.solutionArn());
               System.out.println("The solution name is: " + solution.name());
           }

       } catch (PersonalizeException e) {
           System.err.println(e.awsErrorDetails().errorMessage());
           System.exit(1);
       }
   }
```

- Untuk detail API, lihat ListSolutionsdi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`UpdateCampaign`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`UpdateCampaign`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
    public static String updateCampaign(PersonalizeClient personalizeClient,
            String campaignArn,
            String solutionVersionArn,
            Integer minProvisionedTPS) {

        try {
            // build the updateCampaignRequest
            UpdateCampaignRequest updateCampaignRequest =
UpdateCampaignRequest.builder()
                    .campaignArn(campaignArn)
                    .solutionVersionArn(solutionVersionArn)
                    .minProvisionedTPS(minProvisionedTPS)
                    .build();

            // update the campaign
            personalizeClient.updateCampaign(updateCampaignRequest);

            DescribeCampaignRequest campaignRequest =
DescribeCampaignRequest.builder()
                    .campaignArn(campaignArn)
                    .build();

            DescribeCampaignResponse campaignResponse =
personalizeClient.describeCampaign(campaignRequest);
            Campaign updatedCampaign = campaignResponse.campaign();

            System.out.println("The Campaign status is " +
updatedCampaign.status());
            return updatedCampaign.status();
```

```
        } catch (PersonalizeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return "";
    }
```

- Untuk detail API, lihat UpdateCampaigndi Referensi AWS SDK for Java 2.x API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Contoh kode untuk Amazon Personalize Events menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara menggunakan Amazon Personalize Events dengan AWS software development kit (SDK).

Tindakan merupakan kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Sementara tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks dalam skenario terkait.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Contoh kode

- Contoh dasar untuk Amazon Personalize Events menggunakan AWS SDKs
  - Tindakan untuk Amazon Personalisasi Acara menggunakan AWS SDKs
    - Gunakan PutEvents dengan AWS SDK
    - Gunakan PutItems dengan AWS SDK
    - Gunakan PutUsers dengan AWS SDK

Contoh dasar untuk Amazon Personalize Events menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara menggunakan dasar-dasar Amazon Personalize Events dengan. AWS SDKs

Contoh

- Tindakan untuk Amazon Personalisasi Acara menggunakan AWS SDKs
  - Gunakan PutEvents dengan AWS SDK
  - Gunakan PutItems dengan AWS SDK
  - Gunakan PutUsers dengan AWS SDK

Tindakan untuk Amazon Personalisasi Acara menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara melakukan tindakan Amazon Personalize Events individual dengan. AWS SDKs Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan instruksi untuk mengatur dan menjalankan kode.

Contoh berikut hanya mencakup tindakan yang paling umum digunakan. Untuk daftar lengkapnya, lihat Referensi API Amazon Personalize Events.

Contoh

- Gunakan PutEvents dengan AWS SDK
- Gunakan PutItems dengan AWS SDK
- Gunakan PutUsers dengan AWS SDK

Gunakan **`PutEvents`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`PutEvents`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
       public static int putItems(PersonalizeEventsClient
personalizeEventsClient,
                       String datasetArn,
                       String item1Id,
                       String item1PropertyName,
                       String item1PropertyValue,
                       String item2Id,
                       String item2PropertyName,
                       String item2PropertyValue) {

               int responseCode = 0;
               ArrayList<Item> items = new ArrayList<>();

               try {
                       Item item1 = Item.builder()
                                       .itemId(item1Id)
                                       .properties(String.format("{\"%1$s\":
\"%2$s\"}",
                                                       item1PropertyName,
item1PropertyValue))
                                       .build();

                       items.add(item1);

                       Item item2 = Item.builder()
                                       .itemId(item2Id)
                                       .properties(String.format("{\"%1$s\":
\"%2$s\"}",
                                                       item2PropertyName,
item2PropertyValue))
                                       .build();

                       items.add(item2);

                       PutItemsRequest putItemsRequest =
PutItemsRequest.builder()
                                       .datasetArn(datasetArn)
                                       .items(items)
                                       .build();

                       responseCode =
personalizeEventsClient.putItems(putItemsRequest).sdkHttpResponse().statusCode();
                       System.out.println("Response code: " + responseCode);
```

```
                return responseCode;

            } catch (PersonalizeEventsException e) {
                System.out.println(e.awsErrorDetails().errorMessage());
            }
            return responseCode;
        }
```

- Untuk detail API, lihat PutEventsdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { PutEventsCommand } from "@aws-sdk/client-personalize-events";
import { personalizeEventsClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeEventsClient = new PersonalizeEventsClient({ region:
 "REGION"});

// Convert your UNIX timestamp to a Date.
const sentAtDate = new Date(1613443801 * 1000); // 1613443801 is a testing value.
 Replace it with your sentAt timestamp in UNIX format.

// Set put events parameters.
const putEventsParam = {
  eventList: [
    /* required */
    {
      eventType: "EVENT_TYPE" /* required */,
      sentAt: sentAtDate /* required, must be a Date with js */,
      eventId: "EVENT_ID" /* optional */,
      itemId: "ITEM_ID" /* optional */,
    },
```

```
  ],
  sessionId: "SESSION_ID" /* required */,
  trackingId: "TRACKING_ID" /* required */,
  userId: "USER_ID" /* required */,
};
export const run = async () => {
  try {
    const response = await personalizeEventsClient.send(
      new PutEventsCommand(putEventsParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat PutEventsdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`PutItems`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`PutItems`.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { PutItemsCommand } from "@aws-sdk/client-personalize-events";
```

```
import { personalizeEventsClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeEventsClient = new PersonalizeEventsClient({ region:
 "REGION"});

// Set the put items parameters. For string properties and values, use the \
 character to escape quotes.
const putItemsParam = {
  datasetArn: "DATASET_ARN" /* required */,
  items: [
    /* required */
    {
      itemId: "ITEM_ID" /*  required */,
      properties:
        '{"PROPERTY1_NAME": "PROPERTY1_VALUE", "PROPERTY2_NAME":
 "PROPERTY2_VALUE", "PROPERTY3_NAME": "PROPERTY3_VALUE"}' /* optional */,
    },
  ],
};
export const run = async () => {
  try {
    const response = await personalizeEventsClient.send(
      new PutItemsCommand(putItemsParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat PutItemsdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`PutUsers`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`PutUsers`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
       public static int putUsers(PersonalizeEventsClient
personalizeEventsClient,
                       String datasetArn,
                       String user1Id,
                       String user1PropertyName,
                       String user1PropertyValue,
                       String user2Id,
                       String user2PropertyName,
                       String user2PropertyValue) {

               int responseCode = 0;
               ArrayList<User> users = new ArrayList<>();

               try {
                       User user1 = User.builder()
                                       .userId(user1Id)
                                       .properties(String.format("{\"%1$s\":
\"%2$s\"}",
                                                       user1PropertyName,
user1PropertyValue))
                                       .build();

                       users.add(user1);

                       User user2 = User.builder()
                                       .userId(user2Id)
                                       .properties(String.format("{\"%1$s\":
\"%2$s\"}",
                                                       user2PropertyName,
user2PropertyValue))
                                       .build();
```

```
                        users.add(user2);

                        PutUsersRequest putUsersRequest =
 PutUsersRequest.builder()
                                        .datasetArn(datasetArn)
                                        .users(users)
                                        .build();

                        responseCode =
personalizeEventsClient.putUsers(putUsersRequest).sdkHttpResponse().statusCode();
                        System.out.println("Response code: " + responseCode);
                        return responseCode;

                } catch (PersonalizeEventsException e) {
                        System.out.println(e.awsErrorDetails().errorMessage());
                }
                return responseCode;
        }
```

- Untuk detail API, lihat PutUsersdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { PutUsersCommand } from "@aws-sdk/client-personalize-events";
import { personalizeEventsClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeEventsClient = new PersonalizeEventsClient({ region:
 "REGION"});

// Set the put users parameters. For string properties and values, use the \
 character to escape quotes.
const putUsersParam = {
```

```
  datasetArn: "DATASET_ARN",
  users: [
    {
      userId: "USER_ID",
      properties: '{"PROPERTY1_NAME": "PROPERTY1_VALUE"}',
    },
  ],
};
export const run = async () => {
  try {
    const response = await personalizeEventsClient.send(
      new PutUsersCommand(putUsersParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat PutUsersdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Contoh kode untuk Amazon Personalize Runtime menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara menggunakan Amazon Personalize Runtime dengan AWS perangkat pengembangan perangkat lunak (SDK).

Tindakan merupakan kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Sementara tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks dalam skenario terkait.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Contoh kode

- Contoh dasar untuk Amazon Personalize Runtime menggunakan AWS SDKs
  - Tindakan untuk Amazon Personalisasi Runtime menggunakan AWS SDKs
    - Gunakan GetPersonalizedRanking dengan AWS SDK
    - Gunakan GetRecommendations dengan AWS SDK

Contoh dasar untuk Amazon Personalize Runtime menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara menggunakan dasar-dasar Amazon Personalize Runtime dengan. AWS SDKs

Contoh

- Tindakan untuk Amazon Personalisasi Runtime menggunakan AWS SDKs
  - Gunakan GetPersonalizedRanking dengan AWS SDK
  - Gunakan GetRecommendations dengan AWS SDK

Tindakan untuk Amazon Personalisasi Runtime menggunakan AWS SDKs

Contoh kode berikut menunjukkan cara melakukan tindakan Amazon Personalize Runtime individual dengan. AWS SDKs Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan instruksi untuk mengatur dan menjalankan kode.

Contoh berikut hanya mencakup tindakan yang paling umum digunakan. Untuk daftar lengkapnya, lihat Referensi API Amazon Personalize Runtime.

Contoh

- Gunakan GetPersonalizedRanking dengan AWS SDK
- Gunakan GetRecommendations dengan AWS SDK

Gunakan **GetPersonalizedRanking** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`GetPersonalizedRanking`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
   public static List<PredictedItem> getRankedRecs(PersonalizeRuntimeClient
personalizeRuntimeClient,
           String campaignArn,
           String userId,
           ArrayList<String> items) {

       try {
           GetPersonalizedRankingRequest rankingRecommendationsRequest =
GetPersonalizedRankingRequest.builder()
                   .campaignArn(campaignArn)
                   .userId(userId)
                   .inputList(items)
                   .build();

           GetPersonalizedRankingResponse recommendationsResponse =
personalizeRuntimeClient
                   .getPersonalizedRanking(rankingRecommendationsRequest);
           List<PredictedItem> rankedItems =
recommendationsResponse.personalizedRanking();
           int rank = 1;
           for (PredictedItem item : rankedItems) {
               System.out.println("Item ranked at position " + rank + "
details");
               System.out.println("Item Id is : " + item.itemId());
               System.out.println("Item score is : " + item.score());

System.out.println("-----------------------------------------");
               rank++;
           }
```

```
            return rankedItems;
        } catch (PersonalizeRuntimeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
        return null;
    }
```

- Untuk detail API, lihat GetPersonalizedRankingdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { GetPersonalizedRankingCommand } from "@aws-sdk/client-personalize-
runtime";
import { personalizeRuntimeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region:
 "REGION"});

// Set the ranking request parameters.
export const getPersonalizedRankingParam = {
  campaignArn: "CAMPAIGN_ARN" /* required */,
  userId: "USER_ID" /* required */,
  inputList: ["ITEM_ID_1", "ITEM_ID_2", "ITEM_ID_3", "ITEM_ID_4"],
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(
      new GetPersonalizedRankingCommand(getPersonalizedRankingParam),
    );
    console.log("Success!", response);
```

```
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat GetPersonalizedRankingdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **`GetRecommendations`** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan`GetRecommendations`.

Java

SDK untuk Java 2.x

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

Dapatkan daftar item yang direkomendasikan.

```
   public static void getRecs(PersonalizeRuntimeClient personalizeRuntimeClient,
String campaignArn, String userId) {

       try {
           GetRecommendationsRequest recommendationsRequest =
GetRecommendationsRequest.builder()
                   .campaignArn(campaignArn)
                   .numResults(20)
                   .userId(userId)
                   .build();
```

```
            GetRecommendationsResponse recommendationsResponse =
personalizeRuntimeClient
                    .getRecommendations(recommendationsRequest);
            List<PredictedItem> items = recommendationsResponse.itemList();
            for (PredictedItem item : items) {
                System.out.println("Item Id is : " + item.itemId());
                System.out.println("Item score is : " + item.score());
            }

        } catch (AwsServiceException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

Dapatkan daftar item yang direkomendasikan dari pemberi rekomendasi yang dibuat dalam grup kumpulan data domain.

```
    public static void getRecs(PersonalizeRuntimeClient personalizeRuntimeClient,
String recommenderArn,
            String userId) {

        try {
            GetRecommendationsRequest recommendationsRequest =
GetRecommendationsRequest.builder()
                    .recommenderArn(recommenderArn)
                    .numResults(20)
                    .userId(userId)
                    .build();

            GetRecommendationsResponse recommendationsResponse =
personalizeRuntimeClient
                    .getRecommendations(recommendationsRequest);
            List<PredictedItem> items = recommendationsResponse.itemList();

            for (PredictedItem item : items) {
                System.out.println("Item Id is : " + item.itemId());
                System.out.println("Item score is : " + item.score());
            }
        } catch (AwsServiceException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
```

```
            System.exit(1);
        }
    }
```

Gunakan filter saat meminta rekomendasi.

```
   public static void getFilteredRecs(PersonalizeRuntimeClient
personalizeRuntimeClient,
            String campaignArn,
            String userId,
            String filterArn,
            String parameter1Name,
            String parameter1Value1,
            String parameter1Value2,
            String parameter2Name,
            String parameter2Value) {

        try {

            Map<String, String> filterValues = new HashMap<>();

            filterValues.put(parameter1Name, String.format("\"%1$s\",\"%2$s\"",
                    parameter1Value1, parameter1Value2));
            filterValues.put(parameter2Name, String.format("\"%1$s\"",
                    parameter2Value));

            GetRecommendationsRequest recommendationsRequest =
GetRecommendationsRequest.builder()
                    .campaignArn(campaignArn)
                    .numResults(20)
                    .userId(userId)
                    .filterArn(filterArn)
                    .filterValues(filterValues)
                    .build();

            GetRecommendationsResponse recommendationsResponse =
personalizeRuntimeClient
                    .getRecommendations(recommendationsRequest);
            List<PredictedItem> items = recommendationsResponse.itemList();

            for (PredictedItem item : items) {
                System.out.println("Item Id is : " + item.itemId());
```

```
                System.out.println("Item score is : " + item.score());
            }
        } catch (PersonalizeRuntimeException e) {
            System.err.println(e.awsErrorDetails().errorMessage());
            System.exit(1);
        }
    }
```

- Untuk detail API, lihat GetRecommendationsdi Referensi AWS SDK for Java 2.x API.

JavaScript

SDK untuk JavaScript (v3)

> **Note**
>
> Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di Repositori Contoh Kode AWS.

```
// Get service clients module and commands using ES6 syntax.
import { GetRecommendationsCommand } from "@aws-sdk/client-personalize-runtime";

import { personalizeRuntimeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region:
 "REGION"});

// Set the recommendation request parameters.
export const getRecommendationsParam = {
  campaignArn: "CAMPAIGN_ARN" /* required */,
  userId: "USER_ID" /* required */,
  numResults: 15 /* optional */,
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(
      new GetRecommendationsCommand(getRecommendationsParam),
    );
    console.log("Success!", response);
```

```
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Dapatkan rekomendasi dengan filter (grup kumpulan data khusus).

```
// Get service clients module and commands using ES6 syntax.
import { GetRecommendationsCommand } from "@aws-sdk/client-personalize-runtime";
import { personalizeRuntimeClient } from "./libs/personalizeClients.js";
// Or, create the client here.
// const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region:
 "REGION"});

// Set the recommendation request parameters.
export const getRecommendationsParam = {
  recommenderArn: "RECOMMENDER_ARN" /* required */,
  userId: "USER_ID" /* required */,
  numResults: 15 /* optional */,
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(
      new GetRecommendationsCommand(getRecommendationsParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Dapatkan rekomendasi yang difilter dari pemberi rekomendasi yang dibuat dalam grup kumpulan data domain.

```
// Get service clients module and commands using ES6 syntax.
```

```
import { GetRecommendationsCommand } from "@aws-sdk/client-personalize-runtime";
import { personalizeRuntimeClient } from "./libs/personalizeClients.js";
// Or, create the client here:
// const personalizeRuntimeClient = new PersonalizeRuntimeClient({ region:
 "REGION"});

// Set recommendation request parameters.
export const getRecommendationsParam = {
  campaignArn: "CAMPAIGN_ARN" /* required */,
  userId: "USER_ID" /* required */,
  numResults: 15 /* optional */,
  filterArn: "FILTER_ARN" /* required to filter recommendations */,
  filterValues: {
    PROPERTY:
      '"VALUE"' /* Only required if your filter has a placeholder parameter */,
  },
};

export const run = async () => {
  try {
    const response = await personalizeRuntimeClient.send(
      new GetRecommendationsCommand(getRecommendationsParam),
    );
    console.log("Success!", response);
    return response; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Untuk detail API, lihat GetRecommendationsdi Referensi AWS SDK untuk JavaScript API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihatMenggunakan Amazon Personalize dengan SDK AWS. Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Keamanan di Amazon Personalisasi

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Model tanggung jawab bersama menjelaskan hal ini sebagai keamanan dari cloud dan keamanan dalam cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Amazon Personalize menggunakan enkripsi data untuk melindungi data Anda. Untuk informasi selengkapnya, lihat Enkripsi data di Amazon Personalisasi. Auditor pihak ketiga secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari Program AWS Kepatuhan Program AWS Kepatuhan . Untuk mempelajari tentang program kepatuhan yang berlaku untuk Amazon Personalisasi, lihat AWS Layanan dalam Lingkup berdasarkan AWS Layanan Program Kepatuhan dalam Lingkup oleh Program .
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain, yang mencakup sensitivitas data Anda, persyaratan perusahaan Anda, serta undang-undang dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan Amazon Personalize. Topik berikut menunjukkan cara mengonfigurasi Amazon Personalize untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan sumber daya Amazon Personalisasi Anda.

Topik

- Perlindungan data di Amazon Personalisasi
- Identity and Access Management untuk Amazon Personalize
- Memantau Amazon Personalisasi dengan Amazon CloudWatch
- Logging Amazon Personalisasi panggilan API dengan AWS CloudTrail
- Validasi kepatuhan untuk Amazon Personalize
- Ketahanan di Amazon Personalisasi
- Keamanan infrastruktur di Amazon Personalisasi

- Amazon Personalisasi dan antarmuka titik akhir VPC ()AWS PrivateLink

Perlindungan data di Amazon Personalisasi

Model tanggung jawab AWS bersama model berlaku untuk perlindungan data di Amazon Personalize. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS .

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredensil dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang penggunaan CloudTrail jejak untuk menangkap AWS aktivitas, lihat Bekerja dengan CloudTrail jejak di AWS CloudTrail Panduan Pengguna.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola tingkat lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-3 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi selengkapnya tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-3.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan Amazon Personalize atau lainnya Layanan

AWS menggunakan konsol, API AWS CLI, atau. AWS SDKs Data apa pun yang Anda masukkan ke dalam tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log penagihan atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

Enkripsi data di Amazon Personalisasi

Informasi berikut menjelaskan di mana Amazon Personalize menggunakan enkripsi data untuk melindungi data Anda.

Enkripsi diam

Setiap data yang disimpan dalam Amazon Personalize selalu dienkripsi saat istirahat dengan kunci Amazon Personalize managed (). AWS Key Management Service AWS KMS Jika Anda memberikan AWS KMS kunci Anda sendiri selama pembuatan sumber daya, Amazon Personalize menggunakan kunci untuk mengenkripsi data Anda dan menyimpannya. Misalnya, jika Anda memberikan AWS KMS ARN dalam CreateDatasetGroup operasi, Amazon Personalize menggunakan kunci untuk mengenkripsi dan menyimpan data yang Anda impor ke kumpulan data apa pun yang Anda buat di grup kumpulan data tersebut.

Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.

Untuk informasi tentang enkripsi data di Amazon S3, lihat Melindungi data menggunakan enkripsi di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. Untuk informasi tentang mengelola AWS KMS kunci Anda sendiri, lihat Mengelola kunci di Panduan AWS Key Management Service Pengembang.

Enkripsi bergerak

Amazon Personalize menggunakan TLS dengan AWS sertifikat untuk mengenkripsi data apa pun yang dikirim ke layanan lain. AWS Setiap komunikasi dengan AWS layanan lain terjadi melalui HTTPS, dan titik akhir Amazon Personalize hanya mendukung koneksi aman melalui HTTPS.

Amazon Personalisasi menyalin data dari akun Anda dan memprosesnya dalam sistem internal AWS . Saat memproses data, Amazon Personalize mengenkripsi data dengan kunci Amazon Personalize atau AWS KMS kunci apa pun yang Anda berikan. AWS KMS

Manajemen kunci

AWS mengelola AWS KMS kunci default apa pun. Adalah tanggung jawab Anda untuk mengelola AWS KMS kunci apa pun yang Anda miliki. Anda harus memberikan izin Amazon Personalize dan Amazon Personalize IAM service role untuk menggunakan kunci Anda. Untuk informasi selengkapnya, lihat Memberikan izin Amazon Personalize untuk menggunakan kunci Anda AWS KMS.

Untuk informasi tentang mengelola AWS KMS kunci Anda sendiri, lihat Mengelola kunci di Panduan AWS Key Management Service Pengembang.

Identity and Access Management untuk Amazon Personalize

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diotorisasi (memiliki izin) untuk menggunakan sumber daya Amazon Personalisasi. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

- Audiens
- Mengautentikasi dengan identitas
- Mengelola akses menggunakan kebijakan
- Cara Amazon Personalize bekerja dengan IAM
- Pencegahan "confused deputy" lintas layanan
- Contoh kebijakan berbasis identitas untuk Amazon Personalize
- Memecahkan masalah Amazon Personalisasi identitas dan akses

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan di Amazon Personalize.

Pengguna layanan - Jika Anda menggunakan layanan Amazon Personalize untuk melakukan pekerjaan Anda, administrator Anda akan memberi Anda kredensi dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak fitur Amazon Personalize untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara akses dikelola dapat membantu Anda

meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di Amazon Personalize, lihat. Memecahkan masalah Amazon Personalisasi identitas dan akses

Administrator layanan - Jika Anda bertanggung jawab atas Amazon Personalisasi sumber daya di perusahaan Anda, Anda mungkin memiliki akses penuh ke Amazon Personalize. Tugas Anda adalah menentukan fitur dan sumber daya Amazon Personalisasi mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari selengkapnya tentang cara perusahaan Anda dapat menggunakan IAM dengan Amazon Personalize, lihat. Cara Amazon Personalize bekerja dengan IAM

Administrator IAM - Jika Anda administrator IAM, Anda mungkin ingin mempelajari detail tentang cara menulis kebijakan untuk mengelola akses ke Amazon Personalisasi. Untuk melihat contoh Amazon Personalisasi kebijakan berbasis identitas yang dapat Anda gunakan di IAM, lihat. Contoh kebijakan berbasis identitas untuk Amazon Personalize

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk AWS sebagai identitas federasi dengan menggunakan kredensyal yang disediakan melalui sumber identitas. AWS IAM Identity Center Pengguna (IAM Identity Center), autentikasi masuk tunggal perusahaan Anda, dan kredensi Google atau Facebook Anda adalah contoh identitas federasi. Saat Anda masuk sebagai identitas terfederasi, administrator Anda sebelumnya menyiapkan federasi identitas menggunakan peran IAM. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil peran.

Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal AWS akses. Untuk informasi selengkapnya tentang masuk AWS, lihat Cara masuk ke Panduan AWS Sign-In Pengguna Anda Akun AWS.

Jika Anda mengakses AWS secara terprogram, AWS sediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensil Anda. Jika Anda tidak menggunakan AWS alat, Anda harus menandatangani permintaan sendiri. Guna mengetahui informasi selengkapnya tentang penggunaan metode yang disarankan untuk menandatangani permintaan sendiri, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Misalnya, AWS merekomendasikan agar Anda menggunakan otentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari selengkapnya, lihat Autentikasi multi-faktor dalam Panduan Pengguna AWS IAM Identity Center dan Autentikasi multi-faktor AWS di IAM dalam Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya di akun. Identitas ini disebut pengguna Akun AWS root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Identitas gabungan

Sebagai praktik terbaik, mewajibkan pengguna manusia, termasuk pengguna yang memerlukan akses administrator, untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS dengan menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori pengguna perusahaan Anda, penyedia identitas web, direktori Pusat Identitas AWS Directory Service, atau pengguna mana pun yang mengakses Layanan AWS dengan menggunakan kredensil yang disediakan melalui sumber identitas. Ketika identitas federasi mengakses Akun AWS, mereka mengambil peran, dan peran memberikan kredensyal sementara.

Untuk manajemen akses terpusat, kami sarankan Anda menggunakan AWS IAM Identity Center. Anda dapat membuat pengguna dan grup di Pusat Identitas IAM, atau Anda dapat menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua aplikasi Akun AWS dan aplikasi Anda. Untuk informasi tentang Pusat Identitas IAM, lihat Apakah itu Pusat Identitas IAM? dalam Panduan Pengguna AWS IAM Identity Center .

Pengguna dan grup IAM

Pengguna IAM adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, kami merekomendasikan untuk mengandalkan kredensial

sementara, bukan membuat pengguna IAM yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan tertentu yang memerlukan kredensial jangka panjang dengan pengguna IAM, kami merekomendasikan Anda merotasi kunci akses. Untuk informasi selengkapnya, lihat Merotasi kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensial jangka panjang dalam Panduan Pengguna IAM.

Grup IAM adalah identitas yang menentukan sekumpulan pengguna IAM. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin bagi beberapa pengguna sekaligus. Grup mempermudah manajemen izin untuk sejumlah besar pengguna sekaligus. Misalnya, Anda dapat meminta kelompok untuk menyebutkan IAMAdmins dan memberikan izin kepada grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari selengkapnya, lihat Kasus penggunaan untuk pengguna IAM dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Untuk mengambil peran IAM sementara AWS Management Console, Anda dapat beralih dari pengguna ke peran IAM (konsol). Anda dapat mengambil peran dengan memanggil operasi AWS CLI atau AWS API atau dengan menggunakan URL kustom. Untuk informasi selengkapnya tentang cara menggunakan peran, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM dengan kredensial sementara berguna dalam situasi berikut:

- Akses pengguna terfederasi – Untuk menetapkan izin ke identitas terfederasi, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberi izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat Buat peran untuk penyedia identitas pihak ketiga dalam Panduan Pengguna IAM. Jika menggunakan Pusat Identitas IAM, Anda harus mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM akan mengorelasikan set izin ke peran dalam IAM. Untuk informasi tentang set izin, lihat Set izin dalam Panduan Pengguna AWS IAM Identity Center .
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM guna mendapatkan berbagai izin secara sementara untuk tugas tertentu.

- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (prinsipal tepercaya) di akun lain untuk mengakses sumber daya di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, dengan beberapa Layanan AWS, Anda dapat melampirkan kebijakan secara langsung ke sumber daya (alih-alih menggunakan peran sebagai proxy). Untuk mempelajari perbedaan antara peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.
- Akses lintas layanan — Beberapa Layanan AWS menggunakan fitur lain Layanan AWS. Misalnya, saat Anda melakukan panggilan dalam suatu layanan, biasanya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Sebuah layanan mungkin melakukannya menggunakan izin prinsipal yang memanggil, menggunakan peran layanan, atau peran terkait layanan.
  - Sesi akses teruskan (FAS) — Saat Anda menggunakan pengguna atau peran IAM untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses maju.
  - Peran layanan – Peran layanan adalah peran IAM yang dijalankan oleh layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.
  - Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke peran layanan. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan peran IAM untuk mengelola kredensi sementara untuk aplikasi yang berjalan pada EC2 instance dan membuat AWS CLI atau AWS permintaan API. Ini lebih baik untuk menyimpan kunci akses dalam EC2 instance. Untuk menetapkan AWS peran ke EC2 instance dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instance yang dilampirkan ke instance. Profil instance berisi peran dan memungkinkan program yang berjalan pada EC2 instance untuk mendapatkan kredensi

sementara. Untuk informasi selengkapnya, lihat Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di EC2 instans Amazon di Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan adalah objek AWS yang, ketika dikaitkan dengan identitas atau sumber daya, menentukan izinnya. AWS mengevaluasi kebijakan ini ketika prinsipal (pengguna, pengguna root, atau sesi peran) membuat permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang struktur dan isi dokumen kebijakan JSON, lihat Gambaran umum kebijakan JSON dalam Panduan Pengguna IAM.

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat mengambil peran.

Kebijakan IAM mendefinisikan izin untuk suatu tindakan terlepas dari metode yang Anda gunakan untuk melakukan operasinya. Misalnya, anggaplah Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut bisa mendapatkan informasi peran dari AWS Management Console, API AWS CLI, atau AWS API.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan yang dikelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan

terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran dalam. Akun AWS Kebijakan AWS terkelola mencakup kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan yang dikelola atau kebijakan inline, lihat Pilih antara kebijakan yang dikelola dan kebijakan inline dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan prinsipal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

Daftar kontrol akses (ACLs)

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung. ACLs Untuk mempelajari selengkapnya ACLs, lihat Ringkasan daftar kontrol akses (ACL) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Jenis-jenis kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda oleh jenis kebijakan yang lebih umum.

- Batasan izin – Batasan izin adalah fitur lanjutan tempat Anda mengatur izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas ke entitas IAM (pengguna IAM atau peran IAM). Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian

izin. Untuk informasi selengkapnya tentang batasan izin, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.

- Kebijakan kontrol layanan (SCPs) — SCPs adalah kebijakan JSON yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di. AWS Organizations AWS Organizations adalah layanan untuk mengelompokkan dan mengelola secara terpusat beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur dalam suatu organisasi, maka Anda dapat menerapkan kebijakan kontrol layanan (SCPs) ke salah satu atau semua akun Anda. SCP membatasi izin untuk entitas di akun anggota, termasuk masing-masing. Pengguna root akun AWS Untuk informasi selengkapnya tentang Organizations dan SCPs, lihat Kebijakan kontrol layanan di Panduan AWS Organizations Pengguna.
- Kebijakan kontrol sumber daya (RCPs) — RCPs adalah kebijakan JSON yang dapat Anda gunakan untuk menetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda tanpa memperbarui kebijakan IAM yang dilampirkan ke setiap sumber daya yang Anda miliki. RCP membatasi izin untuk sumber daya di akun anggota dan dapat memengaruhi izin efektif untuk identitas, termasuk Pengguna root akun AWS, terlepas dari apakah itu milik organisasi Anda. Untuk informasi selengkapnya tentang Organizations dan RCPs, termasuk daftar dukungan Layanan AWS tersebut RCPs, lihat Kebijakan kontrol sumber daya (RCPs) di Panduan AWS Organizations Pengguna.
- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter ketika Anda membuat sesi sementara secara programatis untuk peran atau pengguna terfederasi. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan Pengguna IAM.

Cara Amazon Personalize bekerja dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses ke Amazon Personalize, pelajari fitur IAM yang tersedia untuk digunakan dengan Amazon Personalize.

Fitur IAM yang dapat Anda gunakan dengan Amazon Personalize

Fitur IAM	Amazon Personalisasi dukungan
Kebijakan berbasis identitas	Ya
Kebijakan berbasis sumber daya	Tidak
Tindakan kebijakan	Ya
Sumber daya kebijakan	Ya
kunci-kunci persyaratan kebijakan (spesifik layanan)	Ya
ACLs	Tidak
ABAC (tanda dalam kebijakan)	Ya
Kredensial sementara	Ya
Izin principal	Ya
Peran layanan	Ya
Peran terkait layanan	Tidak

Untuk mendapatkan tampilan tingkat tinggi tentang cara Amazon Personalisasi dan layanan AWS lainnya bekerja dengan sebagian besar fitur IAM, AWS lihat layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

Kebijakan berbasis identitas untuk Amazon Personalize

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan secara spesifik apakah tindakan dan sumber daya diizinkan atau ditolak, serta kondisi yang menjadi dasar dikabulkan atau ditolaknya tindakan tersebut. Anda tidak dapat menentukan secara spesifik prinsipal dalam sebuah kebijakan berbasis identitas karena prinsipal berlaku bagi pengguna atau peran yang melekat kepadanya. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan JSON IAM dalam Panduan Pengguna IAM.

Contoh kebijakan berbasis identitas untuk Amazon Personalize

Untuk melihat contoh kebijakan berbasis identitas Amazon Personalisasi, lihat. Contoh kebijakan berbasis identitas untuk Amazon Personalize

Kebijakan berbasis sumber daya dalam Amazon Personalize

Mendukung kebijakan berbasis sumber daya: Tidak

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan prinsipal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan secara spesifik seluruh akun atau entitas IAM di akun lain sebagai prinsipal dalam kebijakan berbasis sumber daya. Menambahkan prinsipal akun silang ke kebijakan berbasis sumber daya hanya setengah dari membangun hubungan kepercayaan. Ketika prinsipal dan sumber daya berbeda Akun AWS, administrator IAM di akun tepercaya juga harus memberikan izin entitas utama (pengguna atau peran) untuk mengakses sumber daya. Mereka memberikan izin dengan melampirkan kebijakan berbasis identitas kepada entitas. Namun, jika kebijakan berbasis sumber daya memberikan akses ke principal dalam akun yang sama, tidak diperlukan kebijakan berbasis identitas tambahan. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Tindakan kebijakan untuk Amazon Personalisasi

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama dengan operasi AWS API terkait. Ada beberapa pengecualian, misalnya tindakan hanya izin yang tidak memiliki operasi API yang cocok. Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar tindakan Personalisasi Amazon, lihat Tindakan yang ditentukan oleh Amazon Personalisasi di Referensi Otorisasi Layanan.

Tindakan kebijakan di Amazon Personalisasi menggunakan awalan berikut sebelum tindakan:

```
personalize
```

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

```
"Action": [
      "personalize:action1",
      "personalize:action2"
         ]
```

Anda juga dapat menentukan beberapa tindakan menggunakan wildcard (*). Sebagai contoh, untuk menentukan semua tindakan yang dimulai dengan kata `Describe`, sertakan tindakan berikut:

```
"Action": "personalize:Describe*"
```

Untuk melihat contoh kebijakan berbasis identitas Amazon Personalisasi, lihat. Contoh kebijakan berbasis identitas untuk Amazon Personalize

Sumber daya kebijakan untuk Amazon Personalisasi

Mendukung sumber daya kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek yang menjadi target penerapan tindakan. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Anda dapat melakukan ini untuk tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*"
```

Untuk melihat daftar jenis sumber daya Amazon Personalisasi dan jenisnya ARNs, lihat Sumber daya yang ditentukan oleh Amazon Personalisasi di Referensi Otorisasi Layanan. Untuk mempelajari tindakan mana yang dapat Anda tentukan ARN dari setiap sumber daya, lihat Tindakan yang ditentukan oleh Amazon Personalisasi.

Untuk melihat contoh kebijakan berbasis identitas Amazon Personalisasi, lihat. Contoh kebijakan berbasis identitas untuk Amazon Personalize

Kunci kondisi kebijakan untuk Amazon Personalize

Mendukung kunci kondisi kebijakan khusus layanan: Yes

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen `Condition` (atau blok `Condition`) akan memungkinkan Anda menentukan kondisi yang menjadi dasar suatu pernyataan berlaku. Elemen `Condition` bersifat opsional. Anda dapat membuat ekspresi bersyarat yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta.

Jika Anda menentukan beberapa elemen `Condition` dalam sebuah pernyataan, atau beberapa kunci dalam elemen `Condition` tunggal, maka AWS akan mengevaluasinya menggunakan operasi

AND logis. Jika Anda menentukan beberapa nilai untuk satu kunci kondisi, AWS mengevaluasi kondisi menggunakan OR operasi logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Sebagai contoh, Anda dapat memberikan izin kepada pengguna IAM untuk mengakses sumber daya hanya jika izin tersebut mempunyai tanda yang sesuai dengan nama pengguna IAM mereka. Untuk informasi selengkapnya, lihat Elemen kebijakan IAM: variabel dan tanda dalam Panduan Pengguna IAM.

AWS mendukung kunci kondisi global dan kunci kondisi khusus layanan. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci kondisi Amazon Personalisasi, lihat Kunci kondisi untuk Amazon Personalisasi di Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat Tindakan yang ditentukan oleh Amazon Personalisasi.

Untuk melihat contoh kebijakan berbasis identitas Amazon Personalisasi, lihat. Contoh kebijakan berbasis identitas untuk Amazon Personalize

ACLs di Amazon Personalisasi

Mendukung ACLs: Tidak

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

ABAC dengan Amazon Personalisasi

Mendukung ABAC (tanda dalam kebijakan): Ya

Kontrol akses berbasis atribut (ABAC) adalah strategi otorisasi yang menentukan izin berdasarkan atribut. Dalam AWS, atribut ini disebut tag. Anda dapat melampirkan tag ke entitas IAM (pengguna atau peran) dan ke banyak AWS sumber daya. Penandaan ke entitas dan sumber daya adalah langkah pertama dari ABAC. Kemudian rancanglah kebijakan ABAC untuk mengizinkan operasi ketika tanda milik prinsipal cocok dengan tanda yang ada di sumber daya yang ingin diakses.

ABAC sangat berguna di lingkungan yang berkembang dengan cepat dan berguna di situasi saat manajemen kebijakan menjadi rumit.

Untuk mengendalikan akses berdasarkan tanda, berikan informasi tentang tanda di elemen kondisi dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi selengkapnya tentang ABAC, lihat Tentukan izin dengan otorisasi ABAC dalam Panduan Pengguna IAM. Untuk melihat tutorial yang menguraikan langkah-langkah pengaturan ABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang menandai sumber daya Amazon Personalize, lihat. Menandai Amazon Personalisasi sumber daya

Untuk melihat contoh kebijakan berbasis identitas untuk membatasi akses ke sumber daya berdasarkan tag pada sumber daya tersebut, lihat Menggunakan tag dalam kebijakan IAM.

Menggunakan kredensi sementara dengan Amazon Personalize

Mendukung kredensial sementara: Ya

Beberapa Layanan AWS tidak berfungsi saat Anda masuk menggunakan kredensi sementara. Untuk informasi tambahan, termasuk yang Layanan AWS bekerja dengan kredensi sementara, lihat Layanan AWS yang bekerja dengan IAM di Panduan Pengguna IAM.

Anda menggunakan kredensyal sementara jika Anda masuk AWS Management Console menggunakan metode apa pun kecuali nama pengguna dan kata sandi. Misalnya, ketika Anda mengakses AWS menggunakan tautan masuk tunggal (SSO) perusahaan Anda, proses tersebut secara otomatis membuat kredensil sementara. Anda juga akan secara otomatis membuat kredensial sementara ketika Anda masuk ke konsol sebagai seorang pengguna lalu beralih peran. Untuk informasi selengkapnya tentang peralihan peran, lihat Beralih dari pengguna ke peran IAM (konsol) dalam Panduan Pengguna IAM.

Anda dapat membuat kredenal sementara secara manual menggunakan API AWS CLI atau AWS . Anda kemudian dapat menggunakan kredensi sementara tersebut untuk mengakses. AWS AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensyal sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat Kredensial keamanan sementara di IAM.

Izin utama lintas layanan untuk Amazon Personalisasi

Mendukung sesi akses maju (FAS): Ya

Saat Anda menggunakan pengguna atau peran IAM untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses maju.

Peran layanan untuk Amazon Personalisasi

Mendukung peran layanan: Ya

Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

> **Warning**
>
> Mengubah izin untuk peran layanan dapat merusak fungsionalitas Amazon Personalize. Edit peran layanan hanya jika Amazon Personalize memberikan panduan untuk melakukannya.

Peran terkait layanan untuk Amazon Personalisasi

Mendukung peran terkait layanan: Tidak

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Untuk detail tentang pembuatan atau manajemen peran terkait layanan, lihat Layanan AWS yang berfungsi dengan IAM. Cari layanan dalam tabel yang memiliki `Yes` di kolom Peran terkait layanan. Pilih tautan Ya untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Pencegahan "confused deputy" lintas layanan

Masalah "confused deputy" adalah masalah keamanan saat entitas yang tidak memiliki izin untuk melakukan suatu tindakan dapat memaksa entitas yang memilik hak akses lebih tinggi untuk melakukan tindakan tersebut. Pada tahun AWS, peniruan lintas layanan dapat mengakibatkan masalah wakil yang membingungkan. Peniruan identitas lintas layanan dapat terjadi ketika satu layanan (layanan yang dipanggil) memanggil layanan lain (layanan yang dipanggil). Layanan pemanggilan dapat dimanipulasi menggunakan izinnya untuk bertindak pada sumber daya pelanggan lain dengan cara yang seharusnya tidak dilakukannya kecuali bila memiliki izin untuk mengakses. Untuk mencegah hal ini, AWS menyediakan alat yang membantu Anda melindungi data untuk semua layanan dengan principal layanan yang telah diberi akses ke sumber daya di akun Anda.

Sebaiknya gunakan kunci konteks kondisi `aws:SourceAccount`global `aws:SourceArn`dan global dalam kebijakan sumber daya untuk membatasi izin yang diberikan Amazon Personalize layanan lain ke sumber daya.

Untuk mencegah masalah wakil yang membingungkan dalam peran yang diambil oleh Amazon Personalize, dalam kebijakan kepercayaan peran tetapkan nilai `aws:SourceArn` to. `arn:aws:personalize:`*`region`*`:`*`accountNumber`*`:*` Wildcard (*) menerapkan kondisi untuk semua resource Amazon Personalize.

Kebijakan hubungan kepercayaan berikut memberikan Amazon Personalisasi akses ke sumber daya Anda dan menggunakan `aws:SourceArn` kunci konteks kondisi global `aws:SourceAccount` dan untuk mencegah masalah deputi yang membingungkan. Gunakan kebijakan ini saat Anda membuat peran untuk Amazon Personalize ()Membuat peran IAM untuk Amazon Personalize.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "personalize.amazonaws.com"
        ]
      },
```

```
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
        "aws:SourceAccount": "111122223333"
        },
        "ArnLike": {
        "aws:SourceArn": "arn:aws:personalize:us-east-1:444455556666:*"
        }
      }
    }
  ]
}
```

Contoh kebijakan berbasis identitas untuk Amazon Personalize

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya Amazon Personalisasi. Mereka juga tidak dapat melakukan tugas dengan menggunakan AWS Management Console, AWS Command Line Interface (AWS CLI), atau AWS API. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat mengambil peran.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM dengan menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat kebijakan IAM (konsol) di Panduan Pengguna IAM.

Untuk detail tentang tindakan dan jenis sumber daya yang ditentukan oleh Amazon Personalisasi, termasuk format ARNs untuk setiap jenis sumber daya, lihat Tindakan, sumber daya, dan kunci kondisi untuk Amazon Personalisasi di Referensi Otorisasi Layanan.

Topik

- Praktik terbaik kebijakan
- AWS kebijakan terkelola
- Menggunakan konsol Amazon Personalize
- Mengizinkan pengguna melihat izin mereka sendiri
- Mengizinkan akses penuh ke Amazon Personalisasi sumber daya
- Mengizinkan akses hanya-baca ke Amazon Personalisasi sumber daya

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus Amazon Personalisasi sumber daya di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS atau Kebijakan yang dikelola AWS untuk fungsi tugas dalam Panduan Pengguna IAM.
- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk mengajukan izin, lihat Kebijakan dan izin dalam IAM dalam Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti AWS CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Kondisi dalam Panduan Pengguna IAM.
- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat Validasi kebijakan dengan IAM Access Analyzer dalam Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Anda, Akun AWS aktifkan MFA untuk keamanan tambahan.

Untuk meminta MFA ketika operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Amankan akses API dengan MFA dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat Praktik terbaik keamanan di IAM dalam Panduan Pengguna IAM.

AWS kebijakan terkelola

AWS kebijakan terkelola adalah kebijakan yang dibuat dan dikelola oleh AWS. Berikut ini adalah contoh kebijakan AWS terkelola yang mungkin Anda gunakan saat bekerja dengan Amazon Personalize.

AmazonPersonalizeFullAccess Kebijakan

Anda dapat menggunakan `AmazonPersonalizeFullAccess` kebijakan AWS terkelola untuk memberi pengguna izin berikut:

- Akses semua sumber daya Amazon Personalisasi
- Publikasikan dan daftar metrik di Amazon CloudWatch
- Buat daftar, baca, tulis, dan hapus semua objek di bucket Amazon S3 yang berisi `Personalize` atau `personalize` dalam nama bucket
- Lulus peran ke Amazon Personalisasi

`AmazonPersonalizeFullAccess`memberikan izin lebih dari yang diperlukan. Sebaiknya buat kebijakan IAM baru yang hanya memberikan izin yang diperlukan (lihat). Memberikan izin Amazon Personalize untuk mengakses sumber daya Anda

CloudWatchFullAccess

Untuk memberikan izin kepada pengguna Anda untuk memantau Amazon Personalize dengan CloudWatch, lampirkan `CloudWatchFullAccess` kebijakan tersebut ke peran Anda. Untuk informasi selengkapnya, lihat Memantau Amazon Personalisasi dengan Amazon CloudWatch.

`CloudWatchFullAccess`Kebijakan ini bersifat opsional dan memberikan izin untuk tindakan berikut:

- Publikasikan dan daftarkan metrik Amazon Personalisasi di CloudWatch

- Lihat metrik dan statistik metrik.
- Atur alarm berbasis metrik.

Menggunakan konsol Amazon Personalize

Untuk mengakses konsol Amazon Personalize, Anda harus memiliki set izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang sumber daya Amazon Personalisasi di sumber daya Anda. Akun AWS Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana mestinya untuk entitas (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau AWS API. Sebagai gantinya, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang coba mereka lakukan.

Mengizinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau menggunakan API atau secara terprogram. AWS CLI AWS

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
```

```
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Mengizinkan akses penuh ke Amazon Personalisasi sumber daya

Contoh berikut memberi pengguna IAM di AWS akun Anda akses penuh ke semua sumber daya dan tindakan Amazon Personalisasi.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "personalize:*"
            ],
            "Resource": "*"
        }
    ]
}
```

Mengizinkan akses hanya-baca ke Amazon Personalisasi sumber daya

Dalam contoh ini, Anda memberi pengguna IAM di AWS akun akses hanya-baca ke sumber daya Amazon Personalisasi, termasuk kumpulan data Amazon Personalisasi, grup kumpulan data, solusi, dan kampanye.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "personalize:DescribeAlgorithm",
                "personalize:DescribeBatchInferenceJob",
                "personalize:DescribeBatchSegmentJob",
                "personalize:DescribeCampaign",
                "personalize:DescribeDataset",
                "personalize:DescribeDatasetExportJob",
                "personalize:DescribeDatasetGroup",
                "personalize:DescribeDatasetImportJob",
                "personalize:DescribeEventTracker",
                "personalize:DescribeFeatureTransformation",
                "personalize:DescribeFilter",
                "personalize:DescribeRecipe",
                "personalize:DescribeRecommender",
                "personalize:DescribeSchema",
                "personalize:DescribeSolution",
                "personalize:DescribeSolutionVersion",
                "personalize:GetSolutionMetrics",
                "personalize:ListBatchInferenceJobs",
                "personalize:ListBatchSegmentJobs",
                "personalize:ListCampaigns",
                "personalize:ListDatasetExportJobs",
                "personalize:ListDatasetGroups",
                "personalize:ListDatasetImportJobs",
                "personalize:ListDatasets",
                "personalize:ListEventTrackers",
                "personalize:ListFilters",
                "personalize:ListRecipes",
                "personalize:ListRecommenders",
                "personalize:ListSchemas",
                "personalize:ListSolutions",
                "personalize:ListSolutionVersions"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

Memecahkan masalah Amazon Personalisasi identitas dan akses

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan Amazon Personalize dan IAM.

Topik

- Saya tidak berwenang untuk melakukan tindakan di Amazon Personalisasi
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar saya Akun AWS mengakses sumber daya Amazon Personalisasi saya

Saya tidak berwenang untuk melakukan tindakan di Amazon Personalisasi

Jika Anda menerima pesan kesalahan bahwa Anda tidak memiliki otorisasi untuk melakukan tindakan, kebijakan Anda harus diperbarui agar Anda dapat melakukan tindakan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM `mateojackson` mencoba menggunakan konsol untuk melihat detail tentang suatu sumber daya *`my-example-widget`* rekaan, tetapi tidak memiliki izin `personalize:`*`GetWidget`* rekaan.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 personalize:GetWidget on resource: my-example-widget
```

Dalam hal ini, kebijakan untuk pengguna `mateojackson` harus diperbarui untuk mengizinkan akses ke sumber daya *`my-example-widget`* dengan menggunakan tindakan `personalize:`*`GetWidget`*.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran ke Amazon Personalisasi.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di Amazon Personalize. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang di luar saya Akun AWS mengakses sumber daya Amazon Personalisasi saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACLs), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mengetahui apakah Amazon Personalize mendukung fitur ini, lihat. Cara Amazon Personalize bekerja dengan IAM
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (federasi identitas) dalam Panduan Pengguna IAM.

- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM di Panduan Pengguna IAM.

Memantau Amazon Personalisasi dengan Amazon CloudWatch

Dengan Amazon CloudWatch, Anda bisa mendapatkan metrik yang terkait dengan Amazon Personalize. Anda dapat mengatur alarm untuk memberi tahu Anda ketika satu atau beberapa metrik ini berada di luar ambang batas yang ditentukan. Untuk melihat metrik, Anda dapat menggunakan Amazon CloudWatch, Amazon AWS Command Line Interface, atau CloudWatch API.

Topik

- Menggunakan CloudWatch metrik untuk Amazon Personalize
- Mengakses metrik Amazon Personalisasi
- Membuat alarm
- Amazon Personalisasi contoh aplikasi pemantauan tanpa server
- CloudWatch metrik untuk Amazon Personalisasi

Menggunakan CloudWatch metrik untuk Amazon Personalize

Untuk menggunakan metrik, Anda harus menentukan informasi berikut:

- Nama metrik.
- Dimensi metrik. Dimensi adalah pasangan nama-nilai yang membantu Anda mengidentifikasi metrik secara unik.

Anda bisa mendapatkan data pemantauan untuk Amazon Personalize menggunakan AWS Management Console, the AWS CLI, atau API. CloudWatch Anda juga dapat menggunakan CloudWatch API melalui salah satu AWS SDKs atau alat CloudWatch API. Konsol menampilkan serangkaian grafik berdasarkan data mentah dari CloudWatch API. Tergantung kebutuhan, Anda mungkin lebih memilih menggunakan grafik yang ditampilkan di konsol atau diterima dari API.

Daftar berikut menunjukkan beberapa penggunaan umum untuk metrik. Berikut ini adalah saran untuk memulai, bukan daftar komprehensif.

Bagaimana cara saya?	Metrik terkait
Bagaimana cara melacak jumlah peristiwa yang telah direkam?	Pantau `PutEventsRequests` metrik.
Bagaimana saya bisa memantau DatasetImportJob kesalahan?	Gunakan metrik `DatasetImportJobError` .
Bagaimana saya bisa memantau latensi `GetRecommendations` panggilan?	Gunakan metrik `GetRecommendationsLatency` .

Anda harus memiliki CloudWatch izin yang sesuai untuk memantau Amazon Personalize CloudWatch dengan. Untuk informasi selengkapnya, lihat Otentikasi dan kontrol akses untuk Amazon CloudWatch.

Mengakses metrik Amazon Personalisasi

Contoh berikut menunjukkan cara mengakses metrik Amazon Personalize menggunakan CloudWatch konsol, the AWS CLI, dan API. CloudWatch

Cara melihat metrik (konsol)

1. Masuk ke AWS Management Console dan buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Pilih Metrik, pilih tab Semua metrik, lalu pilih. `AWS/Personalize`
3. Pilih dimensi metrik.
4. Pilih metrik yang diinginkan dari daftar, dan pilih periode waktu untuk grafik.

Untuk melihat metrik untuk peristiwa yang diterima selama periode waktu tertentu (CLI)

- Buka AWS CLI dan masukkan perintah berikut:

```
aws cloudwatch get-metric-statistics \
  --metric-name PutEventsRequests \
  --start-time 2019-03-15T00:00:20Z \
  --period 3600 \
  --end-time 2019-03-16T00:00:00Z \
```

```
  --namespace AWS/Personalize \
  --dimensions Name=EventTrackerArn,Value=EventTrackerArn \
  --statistics Sum
```

Contoh ini menunjukkan peristiwa yang diterima untuk pelacak acara ARN yang diberikan selama periode waktu tertentu. Untuk informasi selengkapnya, lihat get-metric-statistics.

Untuk mengakses metrik (CloudWatch API)

- Panggil `GetMetricStatistics`. Untuk informasi selengkapnya, lihat Referensi Amazon CloudWatch API.

Membuat alarm

Anda dapat membuat CloudWatch alarm yang mengirimkan pesan Amazon Simple Notification Service (Amazon SNS) saat alarm berubah status. Alarm mengawasi satu metrik selama suatu periode waktu yang Anda tentukan. Alarm melakukan satu atau lebih tindakan berdasarkan nilai metrik relatif terhadap ambang batas tertentu selama beberapa periode waktu. Tindakannya adalah pemberitahuan yang dikirim ke topik Amazon SNS atau kebijakan. AWS Auto Scaling

Alarm memanggil tindakan untuk perubahan status berkelanjutan saja. CloudWatch alarm tidak memanggil tindakan hanya karena mereka berada dalam keadaan tertentu. Status harus diubah dan dipelihara selama jangka waktu tertentu.

Untuk mengatur alarm (konsol)

1. Masuk ke AWS Management Console dan buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, Pilih Alarm, lalu pilih Buat alarm. Proses ini meluncurkan Wizard Buat Alarm.
3. Pilih Pilih Metrik.
4. Di tab Semua metrik, pilih`AWS/Personalize`.
5. Pilih EventTrackerArn, lalu pilih PutEventsRequestsmetrik.
6. Pilih tab Metrik bergrafik.
7. Untuk Statistik pilih Jumlah.
8. Pilih Pilih Metrik.
9. Isi Nama dan Deskripsi. Untuk Kapan pun, pilih >, lalu masukkan nilai maksimum pilihan Anda.

10. Jika Anda CloudWatch ingin mengirimi Anda email ketika status alarm tercapai, untuk Setiap kali alarm ini:, pilih Status adalah ALARM. Untuk mengirim alarm ke topik Amazon SNS yang sudah ada, untuk Kirim notifikasi ke:, pilih topik SNS yang sudah ada. Untuk menetapkan nama dan alamat email untuk daftar langganan email baru, pilih Daftar baru. CloudWatch menyimpan daftar dan menampilkannya di bidang sehingga Anda dapat menggunakannya untuk mengatur alarm future.

    > **Note**
    >
    > Jika Anda menggunakan Daftar baru untuk membuat topik Amazon SNS baru, alamat email harus diverifikasi sebelum penerima yang dituju menerima pemberitahuan. Amazon SNS hanya mengirim email ketika alarm memasuki status alarm. Jika perubahan status alarm ini terjadi sebelum alamat email diverifikasi, maka penerima yang dituju tidak akan menerima notifikasi.

11. Pilih Buat alarm.

Untuk mengatur alarm (AWS CLI)

- Buka AWS CLI, lalu masukkan perintah berikut. Ubah nilai `alarm-actions` parameter untuk mereferensikan topik Amazon SNS yang sebelumnya Anda buat.

```
aws cloudwatch put-metric-alarm \
    --alarm-name PersonalizeCLI \
    --alarm-description "Alarm when more than 10 events occur" \
    --metric-name PutEventsRequests \
    --namespace  AWS/Personalize \
    --statistic Sum \
    --period 300 \
    --threshold 10 \
    --comparison-operator GreaterThanThreshold \
    --evaluation-periods 1 \
    --unit Count \
    --dimensions Name=EventTrackerArn,Value=EventTrackerArn \
    --alarm-actions SNSTopicArn
```

  Contoh ini menunjukkan cara membuat alarm ketika lebih dari 10 peristiwa terjadi untuk ARN pelacak peristiwa yang diberikan dalam waktu 5 menit. Untuk informasi selengkapnya, lihat put-metric-alarm.

Untuk mengatur alarm (CloudWatch API)

- Panggil `PutMetricAlarm`. Untuk informasi selengkapnya, lihat Referensi Amazon CloudWatch API.

Amazon Personalisasi contoh aplikasi pemantauan tanpa server

Untuk contoh aplikasi yang menambahkan kemampuan pemantauan, peringatan, dan pengoptimalan untuk Amazon Personalize, lihat Amazon Personalize monitor di repositori Amazon Personalize samples.

CloudWatch metrik untuk Amazon Personalisasi

Bagian ini berisi informasi tentang CloudWatch metrik Amazon yang tersedia untuk Amazon Personalize. Untuk informasi selengkapnya, lihat Memantau Amazon Personalisasi dengan Amazon CloudWatch.

Tabel berikut mencantumkan metrik Amazon Personalize. Semua metrik kecuali GetRecommendations dan GetPersonalizedRanking mendukung statistik ini:`Average, Minimum, Maximum, Sum`. GetRecommendations dan GetPersonalizedRanking dukungan `Sum` saja.

Metrik	Deskripsi
`DatasetImportJobRequests`	Jumlah panggilan CreateDatasetImportJob API yang berhasil. Dimensi: `DatasetGroupArn, DatasetArn, DatasetImportJobArn`
`DatasetImportJobError`	Jumlah panggilan `CreateDatasetImportJob` API yang mengakibatkan kesalahan. Dimensi: `DatasetGroupArn, DatasetArn, DatasetImportJobArn`
`DatasetImportJobExecutionTime`	Waktu antara panggilan `CreateDatasetImportJob` API dan penyelesaian (atau kegagalan) operasi.

Metrik	Deskripsi
	Dimensi: `DatasetGroupArn, DatasetArn, DatasetImportJobArn` Unit: Detik
`DatasetSize`	Ukuran data yang diimpor oleh pekerjaan impor dataset. Dimensi: `DatasetGroupArn, DatasetArn, DatasetImportJobArn` Unit: Bita
`SolutionTrainingJobRequests`	Jumlah panggilan CreateSolutionVersion API yang berhasil. Dimensi: `SolutionArn, SolutionVersionArn`
`SolutionTrainingJobError`	Jumlah panggilan `CreateSolutionVersion` API yang mengakibatkan kesalahan. Dimensi: `SolutionArn, SolutionVersionArn`
`SolutionTrainingJobExecutionTime`	Waktu antara panggilan `CreateSolutionVersion` API dan penyelesaian (atau kegagalan) operasi. Dimensi: `SolutionArn, SolutionVersionArn` Unit: Detik
`GetPersonalizedRanking`	Apakah panggilan GetPersonalizedRanking API berhasil. Gunakan `sum` statistik untuk melihat jumlah total panggilan GetPersonalizedRanking API yang berhasil. Metrik ini tidak mendukung statistik lainnya. Dimensi: `CampaignArn`

Metrik	Deskripsi
GetPersonalizedRanking4xxErrors	Jumlah panggilan GetPersonalizedRanking API yang mengembalikan kode respons HTTP 4xx. Dimensi: CampaignArn
GetPersonalizedRanking5xxErrors	Jumlah panggilan GetPersonalizedRanking API yang mengembalikan kode respons HTTP 5xx. Dimensi: CampaignArn
GetPersonalizedRankingLatency	Waktu antara menerima panggilan GetPersonalizedRanking API dan pengiriman rekomendasi (tidak termasuk kesalahan 4xx dan 5xx). Dimensi: CampaignArn Satuan: Milidetik
GetRecommendations	Apakah panggilan GetRecommendations API berhasil. Gunakan sum statistik untuk melihat jumlah total panggilan GetRecommendations API yang berhasil. Metrik ini tidak mendukung statistik lainnya. Dimensi: CampaignArn
GetRecommendations4xxErrors	Jumlah panggilan GetRecommendations API yang mengembalikan kode respons HTTP 4xx. Dimensi: CampaignArn
GetRecommendations5xxKesalahan	Jumlah panggilan GetRecommendations API yang mengembalikan kode respons HTTP 5xx. Dimensi: CampaignArn

Metrik	Deskripsi
GetRecommendationsLatency	Waktu antara menerima panggilan GetRecommendations API dan pengiriman rekomendasi (tidak termasuk kesalahan 4xx dan 5xx). Dimensi: CampaignArn Satuan: Milidetik
PutEventsRequests	Jumlah panggilan PutEvents API yang berhasil. Dimensi: DatasetGroupArn, DatasetArn, EventTrackerArn
PutEvents4xxErrors	Jumlah panggilan PutEvents API yang mengembalikan kode respons HTTP 4xx. Dimensi: DatasetGroupArn, DatasetArn, EventTrackerArn
PutEvents5xxErrors	Jumlah panggilan PutEvents API yang mengembalikan kode respons HTTP 5xx. Dimensi: DatasetGroupArn, DatasetArn, EventTrackerArn
PutEventLatency	Waktu yang dibutuhkan untuk menyelesaikan panggilan PutEvents API (tidak termasuk kesalahan 4xx dan 5xx). Dimensi: DatasetGroupArn, DatasetArn, EventTrackerArn Satuan: Milidetik
PutItemsRequests	Jumlah panggilan PutItems API yang berhasil. Dimensi: DatasetGroupArn, DatasetArn

Metrik	Deskripsi
`PutItems4xxErrors`	Jumlah panggilan `PutItems` API yang mengembalikan kode respons HTTP 4xx. Dimensi: `DatasetGroupArn, DatasetArn`
`PutItems5xxErrors`	Jumlah panggilan `PutItems` API yang mengembalikan kode respons HTTP 5xx. Dimensi: `DatasetGroupArn, DatasetArn`
`PutItemsLatency`	Waktu yang dibutuhkan untuk menyelesaikan panggilan `PutItems` API (tidak termasuk kesalahan 4xx dan 5xx). Dimensi: `DatasetGroupArn, DatasetArn` Satuan: Milidetik
`PutUsersRequests`	Jumlah panggilan PutUsers API yang berhasil. Dimensi: `DatasetGroupArn, DatasetArn`
`PutUsers4xxErrors`	Jumlah panggilan `PutUsers` API yang mengembalikan kode respons HTTP 4xx. Dimensi: `DatasetGroupArn, DatasetArn`
`PutUsers5xxErrors`	Jumlah panggilan `PutUsers` API yang mengembalikan kode respons HTTP 5xx. Dimensi: `DatasetGroupArn, DatasetArn`
`PutUsersLatency`	Waktu yang dibutuhkan untuk menyelesaikan panggilan `PutUsers` API (tidak termasuk kesalahan 4xx dan 5xx). Dimensi: `DatasetGroupArn, DatasetArn` Satuan: Milidetik

Logging Amazon Personalisasi panggilan API dengan AWS CloudTrail

Amazon Personalize terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di Amazon Personalize. CloudTrail menangkap subset panggilan API untuk Amazon Personalize sebagai event, termasuk panggilan dari konsol Amazon Personalize dan dari panggilan kode ke Amazon Personalize. APIs Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk acara untuk Amazon Personalisasi. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat untuk Amazon Personalisasi, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk mempelajari selengkapnya CloudTrail, termasuk cara mengonfigurasi dan mengaktifkannya, lihat Panduan AWS CloudTrail Pengguna.

Amazon Personalisasi informasi di CloudTrail

CloudTrail diaktifkan di AWS akun Anda saat Anda membuat akun. Saat aktivitas peristiwa yang didukung terjadi di Amazon Personalize, aktivitas tersebut direkam dalam suatu CloudTrail peristiwa bersama dengan peristiwa AWS layanan lainnya dalam riwayat Acara. Anda dapat melihat, mencari, dan mengunduh acara terbaru di AWS akun Anda. Untuk informasi selengkapnya, lihat Melihat peristiwa dengan riwayat CloudTrail acara.

Untuk catatan peristiwa yang sedang berlangsung di AWS akun Anda, termasuk acara untuk Amazon Personalisasi, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, ketika Anda membuat jejak di konsol, jejak ini diterapkan ke semua Wilayah. Trail mencatat peristiwa dari semua wilayah di AWS partisi dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat:

- Ikhtisar untuk membuat jejak
- CloudTrail layanan dan integrasi yang didukung
- Mengonfigurasi notifikasi Amazon SNS untuk CloudTrail

- Menerima file CloudTrail log dari beberapa wilayah dan Menerima file CloudTrail log dari beberapa akun

Amazon Personalize mendukung pencatatan setiap tindakan (operasi API) sebagai peristiwa dalam file CloudTrail log. Untuk informasi selengkapnya, lihat Tindakan.

Setiap entri peristiwa atau log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut:

- Baik permintaan tersebut dibuat dengan kredensial pengguna atau root.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna gabungan.
- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi selengkapnya, lihat Elemen userIdentity CloudTrail .

Contoh: Amazon Personalisasi entri file log

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa mewakili permintaan tunggal dari sumber manapun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, sehingga file tersebut tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri CloudTrail log dengan tindakan untuk operasi ListDatasetGroups API. Perhatikan bahwa karena operasi ListDatasetGroups API adalah tindakan yang tidak mengubah status, `responseElements` responsnya adalah nol. Untuk informasi selengkapnya tentang isi CloudTrail catatan, lihat isi CloudTrail rekaman.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "principal-id",
        "arn": "arn:aws:iam::user-arn",
        "accountId": "account-id",
        "accessKeyId": "access-key",
```

```
        "userName": "user-name"
    },
    "eventTime": "2018-11-22T02:18:03Z",
    "eventSource": "personalize.amazonaws.com",
    "eventName": "ListDatasetGroups",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "source-ip-address",
    "userAgent": "aws-cli/1.11.16 Python/2.7.11 Darwin/15.6.0 botocore/1.4.73",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "request-id",
    "eventID": "event-id",
    "eventType": "AwsApiCall",
    "recipientAccountId": "recipient-account-id"
}
```


Validasi kepatuhan untuk Amazon Personalize

Auditor pihak ketiga menilai keamanan dan kepatuhan Amazon Personalize sebagai bagian dari AWS beberapa program kepatuhan. Hal ini mencakup SOC, PCI, HIPAA, dan lainnya.

Untuk daftar AWS layanan dalam lingkup program kepatuhan tertentu, lihat AWS layanan dalam lingkup oleh AWS layanan program kepatuhan . Untuk informasi umum, lihat program AWS kepatuhan program AWS .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh laporan di AWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Amazon Personalize ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, serta undang-undang dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- Panduan memulai cepat keamanan dan kepatuhan Panduan memulai penerapan ini membahas pertimbangan arsitektur dan memberikan langkah-langkah untuk menerapkan lingkungan dasar yang berfokus pada keamanan dan kepatuhan. AWS
- Arsitektur untuk whitepaper keamanan dan kepatuhan HIPAA - Pelajari cara Anda dapat menggunakan AWS untuk menjalankan beban kerja sensitif yang diatur berdasarkan Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan AS (HIPAA).
- AWS sumber daya AWS kepatuhan — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.

- Mengevaluasi sumber daya dengan aturan dalam Panduan AWS Config Pengembang — AWS Config Layanan menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.
- AWS Security Hub— AWS Layanan ini memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalamnya AWS yang membantu Anda memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik.

Ketahanan di Amazon Personalisasi

Infrastruktur AWS global dibangun di sekitar AWS Wilayah dan Zona Ketersediaan. AWS Wilayah menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung dengan latensi rendah, throughput tinggi, dan jaringan yang sangat redundan. Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis melakukan fail over di antara zona tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

Amazon Personalize memanfaatkan infrastruktur AWS global untuk ketahanan data. Saat Anda membuat sumber daya Amazon Personalisasi di suatu AWS Wilayah, Amazon Personalize mengelola ketahanan dan redundansi data sumber daya di beberapa Availability Zone. Untuk daftar AWS wilayah tempat Anda dapat membuat sumber daya Amazon Personalisasi, lihat AWS wilayah dan titik akhir di Referensi Umum Amazon Web Services. Untuk informasi selengkapnya tentang AWS Wilayah dan Availability Zone, lihat infrastruktur AWS global.

Keamanan infrastruktur di Amazon Personalisasi

Sebagai layanan terkelola, Amazon Personalize dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat Keamanan AWS Cloud. Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat Perlindungan Infrastruktur dalam Kerangka Kerja yang AWS Diarsiteksikan dengan Baik Pilar Keamanan.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses Amazon Personalize melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.

- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan prinsipal IAM. Atau Anda dapat menggunakan AWS Security Token Service (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

Amazon Personalisasi dan antarmuka titik akhir VPC ()AWS PrivateLink

Jika Anda menggunakan Amazon Virtual Private Cloud (Amazon VPC) untuk meng-host AWS sumber daya Anda, Anda dapat membuat koneksi pribadi antara VPC dan Amazon Personalize. Koneksi ini memungkinkan Amazon Personalize untuk berkomunikasi dengan sumber daya Anda di VPC Anda tanpa melalui internet publik.

Amazon VPC adalah Layanan AWS yang Anda gunakan untuk meluncurkan AWS sumber daya di cloud pribadi virtual (VPC) atau jaringan virtual yang Anda tentukan. Dengan VPC, Anda memiliki kendali terhadap pengaturan jaringan, seperti rentang alamat IP, subnet, tabel rute, dan pintu masuk jaringan. Dengan titik akhir VPC, AWS jaringan menangani perutean antara VPC Anda dan VPC. Layanan AWS

Untuk menghubungkan VPC Anda ke Amazon Personalize, Anda menentukan titik akhir VPC antarmuka untuk Amazon Personalize. Endpoint antarmuka adalah elastic network interface dengan alamat IP pribadi yang berfungsi sebagai titik masuk untuk lalu lintas yang ditujukan untuk didukung. Layanan AWS Titik akhir menyediakan konektivitas yang andal dan dapat diskalakan ke Amazon Personalize. Itu tidak memerlukan gateway internet, instance terjemahan alamat jaringan (NAT), atau koneksi VPN. Untuk informasi selengkapnya, lihat Apa yang dimaksud dengan Amazon VPC dalam Panduan Pengguna Amazon VPC.

Titik akhir VPC antarmuka diaktifkan oleh. AWS PrivateLink AWS Teknologi ini memungkinkan komunikasi pribadi antara Layanan AWS dengan menggunakan elastic network interface dengan alamat IP pribadi.

> **Note**
>
> Semua titik akhir Amazon Personalisasi Federal Information Processing Standard (FIPS) didukung oleh. AWS PrivateLink

Topik

- Membuat titik akhir VPC antarmuka untuk Amazon Personalisasi
- Membuat kebijakan titik akhir VPC untuk Amazon Personalize

Membuat titik akhir VPC antarmuka untuk Amazon Personalisasi

Anda dapat membuat titik akhir VPC untuk layanan Amazon Personalize dengan konsol VPC Amazon atau (). AWS Command Line Interface AWS CLI Untuk informasi selengkapnya, lihat Mengakses AWS layanan menggunakan titik akhir VPC antarmuka di Panduan Pengguna Amazon VPC.

Untuk membuat titik akhir VPC untuk Amazon Personalize, pilih salah satu dari berikut ini untuk layanan ini. Anda dapat memilih untuk membuat titik akhir ipv4, ipv6, atau dualstack.

- com.amazonaws. *region*.personalisasi
- com.amazonaws. *region*.personalisasi-acara
- com.amazonaws. *region*.personalisasi-runtime

Jika Anda mengaktifkan DNS pribadi untuk titik akhir, Anda dapat membuat permintaan API ke Amazon Personalisasi menggunakan nama DNS default untuk Wilayah, misalnya,. `personalize.us-east-1.api.aws`

Membuat kebijakan titik akhir VPC untuk Amazon Personalize

Anda dapat melampirkan kebijakan titik akhir ke titik akhir VPC yang mengontrol akses ke Amazon Personalize. Kebijakan titik akhir menentukan informasi berikut:

- Prinsipal yang dapat melakukan tindakan.
- Tindakan yang dapat dilakukan.
- Sumber daya yang menjadi target tindakan.

Untuk informasi selengkapnya, lihat Mengontrol Akses ke Layanan dengan titik akhir VPC dalam Panduan Pengguna Amazon VPC.

Contoh: Kebijakan titik akhir VPC yang mengizinkan semua tindakan Amazon Personalisasi dan tindakan PassRole

Saat dilampirkan ke titik akhir, kebijakan ini memberikan akses ke semua tindakan Amazon Personalisasi dan tindakan PassRole.

```
{
    "Statement": [
        {
            "Principal": "*",
            "Effect": "Allow",
            "Action": [
                "personalize:*",
                "iam:PassRole"
            ],
            "Resource": "*"
        }
    ]
}
```

Contoh: Kebijakan titik akhir VPC yang mengizinkan tindakan Amazon Personalisasi ListDatasets

Saat dilampirkan ke titik akhir, kebijakan ini memberikan akses ke tindakan Personalisasi Amazon ListDatasets yang terdaftar.

```
{
    "Statement": [
        {
            "Principal": "*",
            "Effect": "Allow",
            "Action": [
                "personalize:ListDatasets"
            ],
            "Resource": "*"
        }
    ]
}
```

Titik akhir dan kuota Amazon Personalize

Bagian berikut berisi informasi tentang pedoman, kuota, dan titik akhir Amazon Personalisasi. Untuk kuota yang dapat disesuaikan, Anda dapat meminta peningkatan kuota menggunakan konsol Service Quotas. Untuk informasi selengkapnya, lihat Meminta peningkatan kuota.

Topik

- Amazon Personalisasi titik akhir dan wilayah
- Kepatuhan
- Kuota layanan
- Meminta peningkatan kuota

Amazon Personalisasi titik akhir dan wilayah

Untuk daftar titik akhir Amazon Personalisasi menurut wilayah, lihat AWS wilayah dan titik akhir di Referensi Umum Amazon Web Services.

Kepatuhan

Untuk informasi tentang Amazon Personalisasi program kepatuhan, lihat AWS kepatuhan, program AWS kepatuhan, dan AWS layanan dalam cakupan berdasarkan program kepatuhan.

Kuota layanan

AWS Akun Anda memiliki kuota berikut untuk Amazon Personalize.

Sumber Daya	Kuota
Item interactions	
Jumlah minimum interaksi item unik yang diperlukan untuk membuat versi solusi atau pemberi rekomendasi. Untuk solusi khusus, Anda harus memiliki banyak catatan ini setelah pemfilteran berdasarkan jenis acara atau nilai acara sebelum pelatihan.	1000

Sumber Daya	Kuota
Untuk User-Personalization-v 2 dan Personalized-Ranking-v 2 resep, jumlah maksimum interaksi item yang dipertimbangkan oleh model selama pelatihan.	3 miliar
Untuk semua kasus penggunaan domain dan resep khusus selain User-Personalization-v Personalized-Ranking-v 2 atau 2, jumlah maksimum interaksi item yang dipertimbangkan oleh model selama pelatihan.	500 juta (dapat disesuaikan)
Jumlah maksimum jenis peristiwa berbeda yang dikombinasikan dengan jumlah total kolom metadata opsional dalam kumpulan data interaksi Item.	10
Jumlah maksimum kolom metadata, tidak termasuk bidang yang dicadangkan, dalam kumpulan data interaksi Item.	5
Jumlah maksimum karakter untuk data kategoris dan nilai tayangan.	1000
Jumlah maksimum data interaksi item massal per pekerjaan impor kumpulan data dengan mode impor PENUH.	100 GB (meningkat menjadi 1TB dengan peningkatan interaksi Item yang dipertimbangkan oleh model)
Jumlah maksimum data interaksi item massal per pekerjaan impor kumpulan data dengan mode impor INCREMENTAL.	1 GB
Jumlah minimum catatan interaksi item per pekerjaan impor kumpulan data dengan mode impor PENUH atau TAMBAHAN.	1000
Users	

Sumber Daya	Kuota
Jumlah minimum pengguna unik dalam data interaksi item, dengan minimal 2 interaksi item masing-masing, diperlukan untuk membuat pemberi rekomendasi domain atau versi solusi khusus.	25
Persentase minimum dari total pengguna yang harus memiliki minimal 2 interaksi item atau lebih sebelum Anda dapat membuat rekomendasi domain atau versi solusi kustom.	1 persen
Jumlah maksimum bidang metadata untuk kumpulan data Pengguna.	25
Jumlah maksimum karakter untuk nilai data USER_ID.	256
Jumlah maksimum karakter untuk nilai data kategoris.	1000 karakter
Jumlah maksimum data pengguna massal per pekerjaan impor kumpulan data dengan mode impor PENUH.	100 GB
Jumlah maksimum data pengguna massal per pekerjaan impor kumpulan data dengan mode impor INCREMENTAL.	1 GB
Items	
Untuk User-Personalization-v 2 atau Personalized-Ranking-v 2, jumlah maksimum item yang dipertimbangkan oleh model selama pelatihan . Item ini berasal dari kumpulan data interaksi Item dan Item.	5 juta

Sumber Daya	Kuota
Untuk semua kasus penggunaan domain dan resep khusus selain User-Personalization-v Personalized-Ranking-v 2 dan 2, jumlah maksimum item yang dipertimbangkan oleh model selama pelatihan dan menghasilkan rekomendasi.	750.000
Jumlah maksimum bidang metadata untuk kumpulan data Item.	100
Jumlah maksimum karakter untuk nilai data ITEM_ID.	256
Jumlah maksimum karakter untuk nilai data string kategoris dan non-kategoris.	1000 karakter
Jumlah maksimum bidang tekstual untuk kumpulan data Item.	1
Jumlah maksimum karakter untuk nilai data tekstual untuk bahasa Mandarin dan Jepang.	7.000 karakter
Jumlah maksimum karakter untuk nilai data tekstual untuk semua bahasa lainnya.	20.000 karakter
Jumlah maksimum data item massal per pekerjaan impor kumpulan data dengan mode impor BULK.	100 GB
Jumlah maksimum data item massal per pekerjaan impor kumpulan data dengan mode impor INCREMENTAL.	1 GB

Actions

Sumber Daya	Kuota
Jumlah maksimum tindakan yang dipertimbangkan oleh model selama pelatihan dan menghasilkan rekomendasi.	1000
Jumlah maksimum bidang metadata untuk kumpulan data Tindakan.	10
Jumlah maksimum karakter untuk nilai data ACTION_ID.	256
Jumlah maksimum karakter untuk nilai data kategoris.	1000 karakter
Jumlah maksimum data tindakan massal per pekerjaan impor kumpulan data dengan mode impor BULK.	100 GB
Jumlah maksimum data tindakan massal per pekerjaan impor kumpulan data dengan mode impor INCREMENTAL.	1 GB
Action interactions	
Jumlah maksimum interaksi tindakan yang dipertimbangkan oleh model selama pelatihan.	500 juta
Jumlah maksimum kolom metadata, tidak termasuk bidang cadangan, dalam kumpulan data Interaksi aksi.	5
Jumlah maksimum data interaksi massal per pekerjaan impor kumpulan data dengan mode impor PENUH.	100 GB (meningkat menjadi 1TB dengan peningkatan interaksi item Tindakan yang dipertimbangkan oleh model)
Jumlah maksimum data interaksi massal per pekerjaan impor kumpulan data dengan mode impor INCREMENTAL.	1 GB

Sumber Daya	Kuota
Individual record import APIs	
Tingkat maksimum `PutEvents` permintaan per grup kumpulan data.	1000/detik
Jumlah maksimum acara dalam `PutEvents` panggilan.	10
Ukuran maksimum suatu acara.	10 KB
Tingkat maksimum `PutActionInteractions` permintaan per grup kumpulan data.	1000/detik
Jumlah maksimum peristiwa interaksi tindakan dalam `PutActionInteractions` panggilan.	10
Ukuran maksimum dari peristiwa interaksi tindakan.	10 KB
Tingkat maksimum `PutItems` permintaan per grup kumpulan data.	10/detik
Jumlah maksimum item dalam `PutItems` panggilan.	10
Tingkat maksimum `PutUsers` permintaan per grup kumpulan data.	10/detik
Jumlah maksimum pengguna dalam `PutUsers` panggilan.	10
Tingkat maksimum `PutActions` permintaan per grup kumpulan data.	10/detik
Jumlah maksimum pengguna dalam `PutActions` panggilan.	10

Sumber Daya	Kuota
Legacy recipes	
Jumlah maksimum data gabungan untuk kumpulan data Pengguna dan Item untuk resep HRNN-metadata dan HRNN-ColdStart.	5 GB
Jumlah maksimum item start dingin yang didukung resep HRNN-ColdStart untuk melatih model (buat versi solusi).	80000
Jumlah minimum item start dingin yang diperlukan resep HRNN-ColdStart untuk melatih model (buat versi solusi).	100
Filters	
Jumlah total filter per grup dataset.	30 (dapat disesuaikan)
Jumlah maksimum bidang dataset yang berbeda untuk filter.	10
Jumlah total bidang kumpulan data yang berbeda di semua filter dalam grup kumpulan data.	20
Jumlah maksimum interaksi item per pengguna per jenis peristiwa yang dipertimbangkan oleh filter.	100 interaksi (dapat disesuaikan)
Jumlah maksimum interaksi tindakan per pengguna per jenis peristiwa yang dipertimb angkan oleh filter.	300 interaksi aksi (dapat disesuaikan)
GetRecommendations / GetPersonalizedRanking / GetActionRecommendations requests	

Sumber Daya	Kuota
Tarif transaksi maksimum untuk`GetRecommendations` , `GetActionRecommendations` dan `GetPersonalizedRanking` permintaan.	2500/dtk
Jumlah maksimum `GetRecommendations` permintaan per detik per kampanye.	500/detik
Jumlah maksimum `GetActionRecommendations` permintaan per detik per kampanye.	500/detik
Jumlah maksimum `GetPersonalizedRanking` permintaan per detik per kampanye.	500/detik.
Jumlah maksimum kolom metadata per `GetRecommendations` atau `GetPersonalizedRanking` permintaan.	10
Jumlah maksimum hasil rekomendasi untuk `GetRecommendation` permintaan tanpa metadata.	500
Jumlah maksimum hasil rekomendasi untuk `GetRecommendation` permintaan dengan metadata.	50
Jumlah maksimum item untuk peringkat dalam `GetPersonalizedRanking` permintaan tanpa metadata.	500
Jumlah maksimum item untuk peringkat dalam `GetPersonalizedRanking` permintaan dengan metadata.	50
Metric attribution quotas	
Jumlah maksimum metrik untuk atribusi metrik	10

Sumber Daya	Kuota
Jumlah maksimum sumber atribusi peristiwa unik	100
Batch inference jobs	
Jumlah maksimum file input untuk pekerjaan inferensi batch.	1000
Ukuran maksimum input pekerjaan inferensi batch.	1 GB
Jumlah maksimum catatan per file input untuk pekerjaan inferensi batch tanpa tema.	50 juta.
Jumlah maksimum catatan per file input untuk pekerjaan inferensi batch dengan tema.	100
Batch segment jobs	
Jumlah maksimum file input untuk pekerjaan segmen batch.	1000
Ukuran maksimum input pekerjaan segmen batch.	1 GB
Jumlah maksimum kueri per file input untuk resep Item-Affinity.	500
Jumlah maksimum kueri per file input untuk Item-Attribute-Affinity resep.	10
Jumlah maksimum pengguna per segmen	5 juta
Data deletion jobs	
Jumlah maksimum pekerjaan penghapusan data untuk grup kumpulan data dengan status PENDING.	5 (dapat disesuaikan)

Sumber Daya	Kuota
Ukuran total maksimum file input penghapusan data atau file	100 MB

AWS Akun Anda memiliki kuota berikut untuk setiap wilayah.

Sumber Daya	Kuota
Jumlah total skema aktif.	500
Jumlah total grup kumpulan data aktif.	5 (dapat disesuaikan)
Jumlah total pekerjaan impor dataset yang tertunda atau sedang berlangsung.	5
Jumlah total pekerjaan inferensi batch yang tertunda atau sedang berlangsung.	5 (dapat disesuaikan)
Jumlah total pekerjaan segmen batch yang tertunda atau sedang berlangsung.	5
Jumlah total versi solusi yang tertunda atau sedang berlangsung.	20 (dapat disesuaikan)

Setiap grup dataset memiliki kuota berikut.

Sumber Daya	Kuota
Jumlah total solusi aktif.	10 (dapat disesuaikan)
Jumlah total kampanye aktif.	5 (dapat disesuaikan)
Jumlah total pemberi rekomendasi.	5

Sumber Daya	Kuota
Jumlah total filter.	30 (dapat disesuaikan)
Jumlah total bidang dataset yang berbeda di semua filter.	20
Jumlah total pekerjaan penghapusan data untuk grup kumpulan data dengan status PENDING.	5

Meminta peningkatan kuota

Untuk kuota yang dapat disesuaikan, Anda dapat meminta peningkatan kuota menggunakan konsol Service Quotas. Kuota Amazon Personalize berikut dapat disesuaikan:

- Jumlah maksimum interaksi item yang dipertimbangkan oleh model selama pelatihan.
- Kampanye aktif per grup kumpulan data
- Grup set data aktif
- Filter aktif per grup kumpulan data
- Solusi aktif per grup dataset
- Jumlah data per impor tambahan
- Jumlah maksimum interaksi item per pengguna per jenis peristiwa yang dipertimbangkan oleh filter
- Jumlah total pekerjaan inferensi batch yang tertunda atau sedang berlangsung
- Jumlah total versi solusi yang tertunda atau sedang berlangsung
- Tingkat maksimum `PutEvents` atau `PutActionInteraction` permintaan

Untuk meminta peningkatan kuota, gunakan konsol Service Quotas dan ikuti langkah-langkah di bagian Meminta peningkatan kuota pada Panduan Pengguna Service Quotas.

Referensi API

Bagian ini menyediakan dokumentasi untuk operasi Amazon Personalize API. Untuk daftar titik akhir Amazon Personalisasi menurut wilayah, lihat AWS wilayah dan titik akhir di Referensi Umum.AWS

Topik

- Tindakan
- Tipe Data
- Kesalahan Umum
- Parameter Umum

Tindakan

Tindakan berikut didukung oleh Amazon Personalize:

- CreateBatchInferenceJob
- CreateBatchSegmentJob
- CreateCampaign
- CreateDataDeletionJob
- CreateDataset
- CreateDatasetExportJob
- CreateDatasetGroup
- CreateDatasetImportJob
- CreateEventTracker
- CreateFilter
- CreateMetricAttribution
- CreateRecommender
- CreateSchema
- CreateSolution
- CreateSolutionVersion
- DeleteCampaign
- DeleteDataset

- DeleteDatasetGroup
- DeleteEventTracker
- DeleteFilter
- DeleteMetricAttribution
- DeleteRecommender
- DeleteSchema
- DeleteSolution
- DescribeAlgorithm
- DescribeBatchInferenceJob
- DescribeBatchSegmentJob
- DescribeCampaign
- DescribeDataDeletionJob
- DescribeDataset
- DescribeDatasetExportJob
- DescribeDatasetGroup
- DescribeDatasetImportJob
- DescribeEventTracker
- DescribeFeatureTransformation
- DescribeFilter
- DescribeMetricAttribution
- DescribeRecipe
- DescribeRecommender
- DescribeSchema
- DescribeSolution
- DescribeSolutionVersion
- GetSolutionMetrics
- ListBatchInferenceJobs
- ListBatchSegmentJobs
- ListCampaigns
- ListDataDeletionJobs

- ListDatasetExportJobs
- ListDatasetGroups
- ListDatasetImportJobs
- ListDatasets
- ListEventTrackers
- ListFilters
- ListMetricAttributionMetrics
- ListMetricAttributions
- ListRecipes
- ListRecommenders
- ListSchemas
- ListSolutions
- ListSolutionVersions
- ListTagsForResource
- StartRecommender
- StopRecommender
- StopSolutionVersionCreation
- TagResource
- UntagResource
- UpdateCampaign
- UpdateDataset
- UpdateMetricAttribution
- UpdateRecommender
- UpdateSolution

Tindakan berikut didukung oleh Amazon Personalize Events:

- PutActionInteractions
- PutActions
- PutEvents
- PutItems

- PutUsers

Tindakan berikut didukung oleh Amazon Personalize Runtime:

- GetActionRecommendations
- GetPersonalizedRanking
- GetRecommendations

Amazon Personalize

Tindakan berikut didukung oleh Amazon Personalize:

- CreateBatchInferenceJob
- CreateBatchSegmentJob
- CreateCampaign
- CreateDataDeletionJob
- CreateDataset
- CreateDatasetExportJob
- CreateDatasetGroup
- CreateDatasetImportJob
- CreateEventTracker
- CreateFilter
- CreateMetricAttribution
- CreateRecommender
- CreateSchema
- CreateSolution
- CreateSolutionVersion
- DeleteCampaign
- DeleteDataset
- DeleteDatasetGroup
- DeleteEventTracker
- DeleteFilter

- DeleteMetricAttribution
- DeleteRecommender
- DeleteSchema
- DeleteSolution
- DescribeAlgorithm
- DescribeBatchInferenceJob
- DescribeBatchSegmentJob
- DescribeCampaign
- DescribeDataDeletionJob
- DescribeDataset
- DescribeDatasetExportJob
- DescribeDatasetGroup
- DescribeDatasetImportJob
- DescribeEventTracker
- DescribeFeatureTransformation
- DescribeFilter
- DescribeMetricAttribution
- DescribeRecipe
- DescribeRecommender
- DescribeSchema
- DescribeSolution
- DescribeSolutionVersion
- GetSolutionMetrics
- ListBatchInferenceJobs
- ListBatchSegmentJobs
- ListCampaigns
- ListDataDeletionJobs
- ListDatasetExportJobs
- ListDatasetGroups
- ListDatasetImportJobs

- ListDatasets
- ListEventTrackers
- ListFilters
- ListMetricAttributionMetrics
- ListMetricAttributions
- ListRecipes
- ListRecommenders
- ListSchemas
- ListSolutions
- ListSolutionVersions
- ListTagsForResource
- StartRecommender
- StopRecommender
- StopSolutionVersionCreation
- TagResource
- UntagResource
- UpdateCampaign
- UpdateDataset
- UpdateMetricAttribution
- UpdateRecommender
- UpdateSolution

CreateBatchInferenceJob

Layanan: Amazon Personalize

Menghasilkan rekomendasi batch berdasarkan daftar item atau pengguna yang disimpan di Amazon S3 dan mengekspor rekomendasi ke bucket Amazon S3.

Untuk menghasilkan rekomendasi batch, tentukan ARN versi solusi dan URI Amazon S3 untuk data input dan output. Untuk personalisasi pengguna, item populer, dan solusi peringkat yang dipersonalisasi, pekerjaan inferensi batch menghasilkan daftar item yang direkomendasikan untuk setiap ID pengguna dalam file input. Untuk solusi item terkait, pekerjaan menghasilkan daftar item yang direkomendasikan untuk setiap ID item dalam file input.

Untuk informasi selengkapnya, lihat Membuat pekerjaan inferensi batch.

Jika Anda menggunakan resep Similar-Items, Amazon Personalize dapat menambahkan tema deskriptif ke rekomendasi batch. Untuk menghasilkan tema, atur mode pekerjaan ke `THEME_GENERATION` dan tentukan nama bidang yang berisi nama item dalam data input.

Untuk informasi selengkapnya tentang membuat tema, lihat Rekomendasi Batch dengan tema dari Content Generator.

Anda tidak bisa mendapatkan rekomendasi batch dengan Trending-Now atau resep. Next-Best-Action

Sintaksis Permintaan

```
{
   "batchInferenceJobConfig": {
      "itemExplorationConfig": {
         "string" : "string"
      }
   },
   "batchInferenceJobMode": "string",
   "filterArn": "string",
   "jobInput": {
      "s3DataSource": {
         "kmsKeyArn": "string",
         "path": "string"
      }
   },
   "jobName": "string",
   "jobOutput": {
```

```
      "s3DataDestination": {
         "kmsKeyArn": "string",
         "path": "string"
      }
   },
   "numResults": number,
   "roleArn": "string",
   "solutionVersionArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ],
   "themeGenerationConfig": {
      "fieldsForThemeGeneration": {
         "itemName": "string"
      }
   }
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

batchInferenceJobConfig

Detail konfigurasi pekerjaan inferensi batch.

Tipe: Objek BatchInferenceJobConfig

Wajib: Tidak

batchInferenceJobMode

Mode pekerjaan inferensi batch. Untuk menghasilkan tema deskriptif untuk grup item serupa, atur mode pekerjaan ke`THEME_GENERATION`. Jika Anda tidak ingin menghasilkan tema, gunakan default`BATCH_INFERENCE`.

Ketika Anda mendapatkan rekomendasi batch dengan tema, Anda akan dikenakan biaya tambahan. Untuk informasi selengkapnya, lihat Harga Amazon Personalisasi.

Tipe: String

Nilai yang Valid: `BATCH_INFERENCE | THEME_GENERATION`

Wajib: Tidak

filterArn

ARN filter untuk diterapkan pada pekerjaan inferensi batch. Untuk informasi selengkapnya tentang penggunaan filter, lihat Memfilter rekomendasi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

jobInput

Jalur Amazon S3 yang mengarah ke file input untuk mendasarkan rekomendasi Anda. Materi input harus dalam format JSON.

Tipe: Objek BatchInferenceJobInput

Wajib: Ya

jobName

Nama pekerjaan inferensi batch yang akan dibuat.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

jobOutput

Jalur ke bucket Amazon S3 tempat output pekerjaan akan disimpan.

Tipe: Objek BatchInferenceJobOutput

Wajib: Ya

numResults

Jumlah rekomendasi untuk diambil.

Tipe: Integer

Wajib: Tidak

roleArn

ARN dari peran Amazon Identity and Access Management yang memiliki izin untuk membaca dan menulis ke bucket Amazon S3 masukan dan keluaran Anda masing-masing.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Ya

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi yang akan digunakan untuk menghasilkan rekomendasi inferensi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke pekerjaan inferensi batch.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

themeGenerationConfig

Untuk pekerjaan pembuatan tema, tentukan nama kolom di kumpulan data Item Anda yang berisi nama setiap item.

Tipe: Objek ThemeGenerationConfig

Wajib: Tidak

Sintaksis Respons

```
{
   "batchInferenceJobArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

batchInferenceJobArn

ARN dari pekerjaan inferensi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateBatchSegmentJob

Layanan: Amazon Personalize

Membuat pekerjaan segmen batch. Operasi dapat menangani hingga 50 juta catatan dan file input harus dalam format JSON. Untuk informasi selengkapnya, lihat Mendapatkan rekomendasi batch dan segmen pengguna.

Sintaksis Permintaan

```
{
   "filterArn": "string",
   "jobInput": {
      "s3DataSource": {
         "kmsKeyArn": "string",
         "path": "string"
      }
   },
   "jobName": "string",
   "jobOutput": {
      "s3DataDestination": {
         "kmsKeyArn": "string",
         "path": "string"
      }
   },
   "numResults": number,
   "roleArn": "string",
   "solutionVersionArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

filterArn

ARN filter untuk diterapkan ke pekerjaan segmen batch. Untuk informasi selengkapnya tentang penggunaan filter, lihat Memfilter rekomendasi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

jobInput

Jalur Amazon S3 untuk data input yang digunakan untuk menghasilkan pekerjaan segmen batch.

Tipe: Objek BatchSegmentJobInput

Wajib: Ya

jobName

Nama pekerjaan segmen batch yang akan dibuat.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

jobOutput

Jalur Amazon S3 untuk bucket tempat output pekerjaan akan disimpan.

Tipe: Objek BatchSegmentJobOutput

Wajib: Ya

numResults

Jumlah pengguna yang diprediksi dihasilkan oleh pekerjaan segmen batch untuk setiap baris data input. Jumlah maksimum pengguna per segmen adalah 5 juta.

Tipe: Integer

Wajib: Tidak

roleArn

ARN dari peran Amazon Identity and Access Management yang memiliki izin untuk membaca dan menulis ke bucket Amazon S3 masukan dan keluaran Anda masing-masing.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Ya

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi yang Anda inginkan untuk digunakan oleh tugas segmen batch untuk menghasilkan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke pekerjaan segmen batch.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "batchSegmentJobArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

batchSegmentJobArn

ARN dari pekerjaan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateCampaign

Layanan: Amazon Personalize

> **Important**
>
> Anda dikenakan biaya kampanye saat sedang aktif. Untuk menghindari biaya yang tidak perlu, pastikan untuk menghapus kampanye ketika Anda selesai. Untuk informasi tentang biaya kampanye, lihat harga Amazon Personalisasi.

Membuat kampanye yang menerapkan versi solusi. Ketika klien memanggil GetRecommendationsdan GetPersonalizedRanking APIs, kampanye ditentukan dalam permintaan.

TPS dan Penskalaan Otomatis yang Disediakan Minimum

> **Important**
>
> Tinggi `minProvisionedTPS` akan meningkatkan biaya Anda. Sebaiknya mulai dengan 1 untuk `minProvisionedTPS` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya.

Saat membuat kampanye Amazon Personalize, Anda dapat menentukan transaksi minimum yang disediakan per detik (`minProvisionedTPS`) untuk kampanye. Ini adalah throughput transaksi dasar untuk kampanye yang disediakan oleh Amazon Personalize. Ini menetapkan biaya penagihan minimum untuk kampanye saat sedang aktif. Transaksi adalah satu `GetRecommendations` atau `GetPersonalizedRanking` permintaan. `minProvisionedTPS`Defaultnya adalah 1.

Jika TPS Anda meningkat lebih dari it`minProvisionedTPS`, Amazon Personalize secara otomatis menskalakan kapasitas yang disediakan ke atas dan ke bawah, tetapi tidak pernah di bawah. `minProvisionedTPS` Ada penundaan waktu singkat sementara kapasitas meningkat yang dapat menyebabkan hilangnya transaksi. Ketika lalu lintas Anda berkurang, kapasitas kembali ke`minProvisionedTPS`.

Anda dikenakan biaya untuk TPS minimum yang disediakan atau, jika permintaan Anda melebihi, TPS yang `minProvisionedTPS` sebenarnya. TPS sebenarnya adalah jumlah permintaan rekomendasi yang Anda buat. Sebaiknya mulai dengan yang rendah`minProvisionedTPS`, lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, lalu tingkatkan `minProvisionedTPS` seperlunya.

Untuk informasi selengkapnya tentang biaya kampanye, lihat harga Amazon Personalisasi.

Status

Kampanye dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Untuk mendapatkan status kampanye, hubungi DescribeCampaign.

> **Note**
>
> Tunggu sampai kampanye `ACTIVE` sebelum meminta rekomendasi kampanye. `status`

Terkait APIs

- ListCampaigns
- DescribeCampaign
- UpdateCampaign
- DeleteCampaign

Sintaksis Permintaan

```
{
   "campaignConfig": {
      "enableMetadataWithRecommendations": boolean,
      "itemExplorationConfig": {
         "string" : "string"
      },
      "syncWithLatestSolutionVersion": boolean
   },
   "minProvisionedTPS": number,
   "name": "string",
   "solutionVersionArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
```

```
        }
    ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

campaignConfig

Detail konfigurasi kampanye.

Tipe: Objek CampaignConfig

Wajib: Tidak

minProvisionedTPS

Menentukan transaksi minimum yang disediakan (rekomendasi) per detik yang akan didukung Amazon Personalize. Tinggi `minProvisionedTPS` akan meningkatkan tagihan Anda. Sebaiknya mulai dengan 1 untuk `minProvisionedTPS` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1.

Wajib: Tidak

name

Nama untuk kampanye baru. Nama kampanye harus unik di akun Anda.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari model terlatih untuk digunakan bersama kampanye. Untuk menentukan versi solusi terbaru dari solusi Anda, tentukan ARN solusi Anda dalam

SolutionArn/$LATEST format. Anda harus menggunakan format ini jika Anda mengatur `syncWithLatestSolutionVersion` ke `True` dalam CampaignConfig.

Untuk menerapkan model yang bukan versi solusi terbaru dari solusi Anda, tentukan ARN versi solusi.

Untuk informasi selengkapnya tentang pembaruan kampanye otomatis, lihat Mengaktifkan pembaruan kampanye otomatis.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke kampanye.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "campaignArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

campaignArn

Amazon Resource Name (ARN) dari kampanye.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateDataDeletionJob

Layanan: Amazon Personalize

Membuat pekerjaan batch yang menghapus semua referensi ke pengguna tertentu dari grup kumpulan data Amazon Personalize dalam batch. Anda menentukan pengguna yang akan dihapus dalam file CSV UserIds di bucket Amazon S3. Setelah pekerjaan selesai, Amazon Personalize tidak lagi melatih data pengguna dan tidak lagi mempertimbangkan pengguna saat membuat segmen pengguna. Untuk informasi selengkapnya tentang membuat pekerjaan penghapusan data, lihat Menghapus pengguna.

- File input Anda harus berupa file CSV dengan kolom USER_ID tunggal yang mencantumkan pengguna. IDs Untuk informasi selengkapnya tentang menyiapkan file CSV, lihat Mempersiapkan file penghapusan data Anda dan mengunggahnya ke Amazon S3.
- Untuk memberikan izin Amazon Personalize untuk mengakses file CSV input userIds, Anda harus menentukan peran layanan IAM yang memiliki izin untuk membaca dari sumber data. Peran ini membutuhkan `GetObject` dan `ListBucket` izin untuk bucket dan kontennya. Izin ini sama dengan mengimpor data. Untuk informasi tentang pemberian akses ke bucket Amazon S3, lihat Memberikan Amazon Personalisasi Akses ke Sumber Daya Amazon S3.

Setelah Anda membuat pekerjaan, dibutuhkan waktu hingga satu hari untuk menghapus semua referensi ke pengguna dari kumpulan data dan model. Sampai pekerjaan selesai, Amazon Personalize terus menggunakan data saat pelatihan. Dan jika Anda menggunakan resep Segmentasi Pengguna, pengguna mungkin muncul di segmen pengguna.

Status

Pekerjaan penghapusan data dapat memiliki salah satu status berikut:

- TERTUNDA > IN_PROGRESS> SELESAI -atau- GAGAL

Untuk mendapatkan status pekerjaan penghapusan data, panggil operasi DescribeDataDeletionJobAPI dan tentukan Nama Sumber Daya Amazon (ARN) pekerjaan. Jika statusnya GAGAL, respons menyertakan `failureReason` kunci, yang menjelaskan mengapa pekerjaan gagal.

Terkait APIs

- ListDataDeletionJobs

- DescribeDataDeletionJob

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "dataSource": {
      "dataLocation": "string"
   },
   "jobName": "string",
   "roleArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data yang memiliki kumpulan data yang ingin Anda hapus catatan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

dataSource

Bucket Amazon S3 yang berisi daftar userIds pengguna yang akan dihapus.

Tipe: Objek DataSource

Wajib: Ya

jobName

Nama untuk pekerjaan penghapusan data.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

roleArn

Nama Sumber Daya Amazon (ARN) dari peran IAM yang memiliki izin untuk membaca dari sumber data Amazon S3.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke pekerjaan penghapusan data.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "dataDeletionJobArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

dataDeletionJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan penghapusan data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateDataset

Layanan: Amazon Personalize

Membuat dataset kosong dan menambahkannya ke grup dataset tertentu. Gunakan CreateDatasetImportJobuntuk mengimpor data pelatihan Anda ke kumpulan data.

Ada 5 jenis dataset:

- Interaksi item
- Item
- Pengguna
- Interaksi aksi
- Tindakan

Setiap jenis kumpulan data memiliki skema terkait dengan jenis bidang yang diperlukan. Hanya `Item interactions` kumpulan data yang diperlukan untuk melatih model (juga disebut sebagai membuat solusi).

Dataset dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Untuk mendapatkan status kumpulan data, hubungi DescribeDataset.

Terkait APIs

- CreateDatasetGroup
- ListDatasets
- DescribeDataset
- DeleteDataset

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "datasetType": "string",
```

```
    "name": "string",
    "schemaArn": "string",
    "tags": [
       {
          "tagKey": "string",
          "tagValue": "string"
       }
    ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data untuk menambahkan kumpulan data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

datasetType

Jenis dataset.

Salah satu nilai berikut (case insensitive):

- Interaksi
- Item
- Pengguna
- Tindakan
- Action_Interactions

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Ya

name

Nama untuk dataset.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

schemaArn

ARN dari skema untuk dikaitkan dengan dataset. Skema mendefinisikan bidang dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke dataset.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "datasetArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetArn

ARN dari dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateDatasetExportJob

Layanan: Amazon Personalize

Membuat pekerjaan yang mengekspor data dari kumpulan data Anda ke bucket Amazon S3. Untuk mengizinkan Amazon Personalize mengekspor data pelatihan, Anda harus menentukan peran IAM terkait layanan yang memberikan izin Amazon Personalize untuk `PutObject` bucket Amazon S3 Anda. Untuk selengkapnya, lihat Mengekspor kumpulan data di panduan developer Amazon Personalize.

Status

Pekerjaan ekspor dataset dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Untuk mendapatkan status pekerjaan ekspor, hubungi DescribeDatasetExportJob, dan tentukan Nama Sumber Daya Amazon (ARN) dari pekerjaan ekspor kumpulan data. Ekspor kumpulan data selesai ketika status ditampilkan sebagai AKTIF. Jika status ditampilkan sebagai CREATE FAILED, respons menyertakan `failureReason` kunci, yang menjelaskan mengapa pekerjaan gagal.

Sintaksis Permintaan

```
{
   "datasetArn": "string",
   "ingestionMode": "string",
   "jobName": "string",
   "jobOutput": {
      "s3DataDestination": {
         "kmsKeyArn": "string",
         "path": "string"
      }
   },
   "roleArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data yang berisi data yang akan diekspor.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

ingestionMode

Data yang akan diekspor, berdasarkan cara Anda mengimpor data. Anda dapat memilih untuk mengekspor hanya `BULK` data yang diimpor menggunakan pekerjaan impor kumpulan data, hanya `PUT` data yang Anda impor secara bertahap (menggunakan konsol PutEvents, PutUsers dan PutItems operasi), atau `ALL` untuk kedua jenis. Nilai default-nya adalah `PUT`.

Tipe: String

Nilai yang Valid: `BULK | PUT | ALL`

Wajib: Tidak

jobName

Nama untuk pekerjaan ekspor dataset.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

jobOutput

Jalur ke bucket Amazon S3 tempat output pekerjaan disimpan.

Tipe: Objek DatasetExportJobOutput

Wajib: Ya

roleArn

Nama Sumber Daya Amazon (ARN) dari peran layanan IAM yang memiliki izin untuk menambahkan data ke bucket Amazon S3 keluaran Anda.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke pekerjaan ekspor dataset.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "datasetExportJobArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetExportJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan ekspor dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateDatasetGroup

Layanan: Amazon Personalize

Membuat grup dataset kosong. Grup kumpulan data adalah wadah untuk sumber daya Amazon Personalize. Grup kumpulan data dapat berisi paling banyak tiga kumpulan data, satu untuk setiap jenis kumpulan data:

- Interaksi item
- Item
- Pengguna
- Tindakan
- Interaksi aksi

Grup kumpulan data dapat berupa grup kumpulan data Domain, tempat Anda menentukan domain dan menggunakan sumber daya yang telah dikonfigurasi sebelumnya seperti pemberi rekomendasi, atau grup kumpulan data khusus, tempat Anda menggunakan sumber daya khusus, seperti solusi dengan versi solusi, yang Anda gunakan dengan kampanye. Jika memulai dengan grup kumpulan data Domain, Anda masih dapat menambahkan sumber daya khusus seperti solusi dan versi solusi yang dilatih dengan resep untuk kasus penggunaan khusus dan diterapkan dengan kampanye.

Grup kumpulan data dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA

Untuk mendapatkan status grup kumpulan data, hubungi DescribeDatasetGroup. Jika status ditampilkan sebagai CREATE FAILED, responsnya menyertakan `failureReason` kunci, yang menjelaskan mengapa pembuatan gagal.

> **Note**
>
> Anda harus menunggu hingga grup dataset `ACTIVE` sebelum menambahkan kumpulan data ke grup. `status`

Anda dapat menentukan kunci AWS Key Management Service (KMS) untuk mengenkripsi kumpulan data dalam grup. Jika Anda menentukan kunci KMS, Anda juga harus menyertakan peran AWS Identity and Access Management (IAM) yang memiliki izin untuk mengakses kunci.

APIs yang membutuhkan grup kumpulan data ARN dalam permintaan

- CreateDataset
- CreateEventTracker
- CreateSolution

Terkait APIs

- ListDatasetGroups
- DescribeDatasetGroup
- DeleteDatasetGroup

Sintaksis Permintaan

```
{
   "domain": "string",
   "kmsKeyArn": "string",
   "name": "string",
   "roleArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

domain

Domain dari grup dataset. Tentukan domain untuk membuat grup dataset Domain. Domain yang Anda tentukan menentukan skema default untuk kumpulan data dan kasus penggunaan yang

tersedia untuk pemberi rekomendasi. Jika Anda tidak menentukan domain, Anda membuat grup kumpulan data kustom dengan versi solusi yang Anda terapkan dengan kampanye.

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Wajib: Tidak

kmsKeyArn

Nama Sumber Daya Amazon (ARN) dari kunci AWS Key Management Service (KMS) yang digunakan untuk mengenkripsi kumpulan data.

Tipe: String

Batasan Panjang: Panjang maksimum 2048.

Pola: `arn:aws.*:kms:.*:[0-9]{12}:key/.*`

Wajib: Tidak

name

Nama untuk grup dataset baru.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

roleArn

ARN dari peran AWS Identity and Access Management (IAM) yang memiliki izin untuk mengakses kunci AWS Key Management Service (KMS). Menyediakan peran IAM hanya valid ketika juga menentukan kunci KMS.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Tidak

tags

Daftar tag untuk diterapkan ke grup dataset.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "datasetGroupArn": "string",
   "domain": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset baru.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

domain

Domain untuk grup dataset Domain baru.

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateDatasetImportJob

Layanan: Amazon Personalize

Membuat job yang mengimpor data pelatihan dari sumber data Anda (bucket Amazon S3) ke kumpulan data Amazon Personalize. Untuk mengizinkan Amazon Personalisasi mengimpor data pelatihan, Anda harus menentukan peran layanan IAM yang memiliki izin untuk membaca dari sumber data, karena Amazon Personalize membuat salinan data Anda dan memprosesnya secara internal. Untuk informasi tentang pemberian akses ke bucket Amazon S3, lihat Memberikan Amazon Personalisasi Akses ke Sumber Daya Amazon S3.

Jika Anda telah membuat pemberi rekomendasi atau menerapkan versi solusi kustom dengan kampanye, bagaimana catatan massal baru memengaruhi rekomendasi bergantung pada kasus penggunaan domain atau resep yang Anda gunakan. Untuk informasi selengkapnya, lihat Bagaimana data baru memengaruhi rekomendasi waktu nyata.

> **Important**
>
> Secara default, pekerjaan impor dataset menggantikan data yang ada dalam kumpulan data yang Anda impor secara massal. Untuk menambahkan catatan baru tanpa mengganti data yang ada, tentukan INCREMENTAL untuk mode impor dalam CreateDatasetImportJob operasi.

Status

Pekerjaan impor dataset dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Untuk mendapatkan status pekerjaan impor, hubungi DescribeDatasetImportJob, berikan Nama Sumber Daya Amazon (ARN) dari pekerjaan impor kumpulan data. Impor dataset selesai ketika status ditampilkan sebagai ACTIVE. Jika status ditampilkan sebagai CREATE FAILED, responsnya menyertakan `failureReason` kunci, yang menjelaskan mengapa pekerjaan gagal.

> **Note**
>
> Mengimpor membutuhkan waktu. Anda harus menunggu hingga status ditampilkan sebagai AKTIF sebelum melatih model menggunakan kumpulan data.

Terkait APIs

- ListDatasetImportJobs
- DescribeDatasetImportJob

Sintaksis Permintaan

```
{
   "datasetArn": "string",
   "dataSource": {
      "dataLocation": "string"
   },
   "importMode": "string",
   "jobName": "string",
   "publishAttributionMetricsToS3": boolean,
   "roleArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

ARN dari dataset yang menerima data yang diimpor.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

dataSource

Bucket Amazon S3 yang berisi data pelatihan untuk diimpor.

Tipe: Objek DataSource

Wajib: Ya

importMode

Tentukan cara menambahkan catatan baru ke kumpulan data yang ada. Mode impor default adalahFULL. Jika Anda belum mengimpor catatan massal ke dalam kumpulan data sebelumnya, Anda hanya dapat menentukanFULL.

- Tentukan FULL untuk menimpa semua data massal yang ada di kumpulan data Anda. Data yang Anda impor satu per satu tidak diganti.
- Tentukan INCREMENTAL untuk menambahkan catatan baru ke data yang ada di kumpulan data Anda. Amazon Personalize menggantikan catatan apa pun dengan ID yang sama dengan yang baru.

Tipe: String

Nilai yang Valid: FULL | INCREMENTAL

Wajib: Tidak

jobName

Nama untuk pekerjaan impor dataset.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: ^[a-zA-Z0-9][a-zA-Z0-9\-_]*

Wajib: Ya

publishAttributionMetricsToS3

Jika Anda membuat atribusi metrik, tentukan apakah akan memublikasikan metrik untuk pekerjaan impor ini ke Amazon S3

Tipe: Boolean

Wajib: Tidak

roleArn

ARN dari peran IAM yang memiliki izin untuk membaca dari sumber data Amazon S3.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke pekerjaan impor dataset.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "datasetImportJobArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetImportJobArn

ARN dari pekerjaan impor dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateEventTracker

Layanan: Amazon Personalize

Membuat pelacak peristiwa yang Anda gunakan saat menambahkan data peristiwa ke grup kumpulan data tertentu menggunakan API. PutEvents

> **Note**
>
> Hanya satu pelacak peristiwa yang dapat dikaitkan dengan grup kumpulan data. Anda akan mendapatkan kesalahan jika Anda memanggil `CreateEventTracker` menggunakan grup kumpulan data yang sama dengan pelacak peristiwa yang ada.

Saat Anda membuat pelacak peristiwa, respons menyertakan ID pelacakan, yang Anda lewati sebagai parameter saat Anda menggunakan PutEventsoperasi. Amazon Personalisasi kemudian menambahkan data peristiwa ke kumpulan data interaksi Item dari grup kumpulan data yang Anda tentukan di pelacak peristiwa.

Pelacak acara dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Untuk mendapatkan status pelacak acara, hubungi DescribeEventTracker.

> **Note**
>
> Pelacak peristiwa harus dalam status AKTIF sebelum menggunakan ID pelacakan.

Terkait APIs

- ListEventTrackers
- DescribeEventTracker
- DeleteEventTracker

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "name": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data yang menerima data peristiwa.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

name

Nama untuk pelacak acara.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke pelacak acara.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "eventTrackerArn": "string",
   "trackingId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

eventTrackerArn

ARN dari pelacak acara.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

trackingId

ID pelacak acara. Sertakan ID ini dalam permintaan ke PutEventsAPI.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript

- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateFilter

Layanan: Amazon Personalize

Membuat filter rekomendasi. Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "filterExpression": "string",
   "name": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

ARN dari grup dataset tempat filter akan menjadi milik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

filterExpression

Ekspresi filter mendefinisikan item mana yang disertakan atau dikecualikan dari rekomendasi. Ekspresi filter harus mengikuti aturan format tertentu. Untuk informasi tentang struktur ekspresi filter dan sintaks, lihat Menyaring ekspresi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 2500.

Wajib: Ya

name

Nama filter yang akan dibuat.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke filter.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "filterArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

filterArn

ARN dari filter baru.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateMetricAttribution

Layanan: Amazon Personalize

Membuat atribusi metrik. Atribusi metrik membuat laporan tentang data yang Anda impor ke Amazon Personalize. Bergantung pada cara Anda mengimpor data, Anda dapat melihat laporan di Amazon CloudWatch atau Amazon S3. Untuk informasi selengkapnya, lihat Mengukur dampak rekomendasi.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "metrics": [
      {
         "eventType": "string",
         "expression": "string",
         "metricName": "string"
      }
   ],
   "metricsOutputConfig": {
      "roleArn": "string",
      "s3DataDestination": {
         "kmsKeyArn": "string",
         "path": "string"
      }
   },
   "name": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data tujuan untuk atribusi metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

metrics

Daftar atribut metrik untuk atribusi metrik. Setiap atribut metrik menentukan jenis acara untuk melacak dan fungsi. Fungsi yang tersedia adalah `SUM()` atau`SAMPLECOUNT()`. Untuk fungsi SUM (), berikan jenis kumpulan data (baik Interaksi atau Item) dan kolom untuk dijumlahkan sebagai parameter. Misalnya SUM (Items.price).

Tipe: Array objek MetricAttribute

Anggota Array: Jumlah maksimum 10 item.

Wajib: Ya

metricsOutputConfig

Rincian konfigurasi keluaran untuk atribusi metrik.

Tipe: Objek MetricAttributionOutput

Wajib: Ya

name

Nama untuk atribusi metrik.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Diperlukan: Ya

Sintaksis Respons

```
{
   "metricAttributionArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

metricAttributionArn

Nama Sumber Daya Amazon (ARN) untuk atribusi metrik baru.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateRecommender

Layanan: Amazon Personalize

Membuat pemberi rekomendasi dengan resep (kasus penggunaan grup kumpulan data Domain) yang Anda tentukan. Anda membuat rekomendasi untuk grup kumpulan data Domain dan menentukan Amazon Resource Name (ARN) pemberi rekomendasi saat Anda membuat permintaan. GetRecommendations

Permintaan rekomendasi minimum per detik

> **Important**
>
> Tinggi `minRecommendationRequestsPerSecond` akan meningkatkan tagihan Anda. Sebaiknya mulai dengan 1 untuk `minRecommendationRequestsPerSecond` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minRecommendationRequestsPerSecond` seperlunya.

Saat membuat pemberi rekomendasi, Anda dapat mengonfigurasi permintaan rekomendasi minimum pemberi rekomendasi per detik. Permintaan rekomendasi minimum per detik (`minRecommendationRequestsPerSecond`) menentukan throughput permintaan rekomendasi dasar yang disediakan oleh Amazon Personalize. minRecommendationRequestsPerSecond Defaultnya adalah1. Permintaan rekomendasi adalah `GetRecommendations` operasi tunggal. Throughput permintaan diukur dalam permintaan per detik dan Amazon Personalize menggunakan permintaan Anda per detik untuk mendapatkan permintaan Anda per jam dan harga penggunaan pemberi rekomendasi Anda.

Jika permintaan per detik Anda meningkat lebih dari itu`minRecommendationRequestsPerSecond`, Amazon Personalize secara otomatis menskalakan kapasitas yang disediakan ke atas dan ke bawah, tetapi tidak pernah di bawah ini. `minRecommendationRequestsPerSecond` Ada penundaan waktu singkat sementara kapasitas meningkat yang dapat menyebabkan hilangnya permintaan.

Tagihan Anda lebih besar dari permintaan minimum per jam (berdasarkan minRecommendationRequestsPerSecond) atau jumlah permintaan yang sebenarnya. Throughput permintaan aktual yang digunakan dihitung sebagai rata-rata requests/second dalam jendela satu jam. Sebaiknya mulai dengan default`minRecommendationRequestsPerSecond`, lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan kemudian tingkatkan `minRecommendationRequestsPerSecond` seperlunya.

Status

Seorang pemberi rekomendasi dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- STOP PENDING > STOP IN_PROGRESS > TIDAK AKTIF> MULAI TERTUNDA > MULAI IN_PROGRESS > AKTIF
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Untuk mendapatkan status pemberi rekomendasi, hubungi DescribeRecommender.

> **Note**
>
> Tunggu sampai pemberi rekomendasi `ACTIVE` sebelum meminta rekomendasi dari pemberi rekomendasi. `status`

Terkait APIs

- ListRecommenders
- DescribeRecommender
- UpdateRecommender
- DeleteRecommender

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "name": "string",
   "recipeArn": "string",
   "recommenderConfig": {
      "enableMetadataWithRecommendations": boolean,
      "itemExplorationConfig": {
         "string" : "string"
      },
      "minRecommendationRequestsPerSecond": number,
      "trainingDataConfig": {
         "excludedDatasetColumns": {
            "string" : [ "string" ]
```

```
            }
         }
      },
      "tags": [
         {
            "tagKey": "string",
            "tagValue": "string"
         }
      ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data domain tujuan untuk pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

name

Nama pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

recipeArn

Nama Sumber Daya Amazon (ARN) dari resep yang akan digunakan oleh pemberi rekomendasi. Untuk pemberi rekomendasi, resep adalah kasus penggunaan grup kumpulan data Domain.

Hanya kasus penggunaan grup kumpulan data Domain yang dapat digunakan untuk membuat pemberi rekomendasi. Untuk informasi tentang kasus penggunaan, lihat Memilih kasus penggunaan pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

recommenderConfig

Detail konfigurasi dari pemberi rekomendasi.

Tipe: Objek RecommenderConfig

Wajib: Tidak

tags

Daftar tag untuk diterapkan ke pemberi rekomendasi.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "recommenderArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateSchema

Layanan: Amazon Personalize

Membuat skema Amazon Personalize dari string skema yang ditentukan. Skema yang Anda buat harus dalam format Avro JSON.

Amazon Personalize mengenali tiga varian skema. Setiap skema dikaitkan dengan jenis dataset dan memiliki satu set bidang dan kata kunci yang diperlukan. Jika Anda membuat skema untuk kumpulan data dalam grup kumpulan data Domain, Anda menyediakan domain grup kumpulan data Domain. Anda menentukan skema saat Anda menelepon CreateDataset.

Untuk informasi lebih lanjut tentang skema, lihat Kumpulan data dan skema.

Terkait APIs

- ListSchemas
- DescribeSchema
- DeleteSchema

Sintaksis Permintaan

```
{
   "domain": "string",
   "name": "string",
   "schema": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

domain

Domain untuk skema. Jika Anda membuat skema untuk kumpulan data dalam grup kumpulan data Domain, tentukan domain yang Anda pilih saat membuat grup kumpulan data Domain.

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Wajib: Tidak

name

Nama untuk skema.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

schema

Skema dalam format Avro JSON.

Tipe: String

Kendala Panjang: Panjang maksimum 20000.

Wajib: Ya

Sintaksis Respons

```
{
   "schemaArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

schemaArn

Nama Sumber Daya Amazon (ARN) dari skema yang dibuat.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateSolution

Layanan: Amazon Personalize

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Membuat konfigurasi untuk melatih model (membuat versi solusi). Konfigurasi ini mencakup resep yang akan digunakan untuk pelatihan model dan konfigurasi pelatihan opsional, seperti kolom yang akan digunakan dalam pelatihan dan parameter transformasi fitur. Untuk informasi selengkapnya tentang mengonfigurasi solusi, lihat Membuat dan mengonfigurasi solusi.

Secara default, solusi baru menggunakan pelatihan otomatis untuk membuat versi solusi setiap 7 hari. Anda dapat mengubah frekuensi pelatihan. Pembuatan versi solusi otomatis dimulai dalam waktu satu jam setelah solusi AKTIF. Jika Anda membuat versi solusi secara manual dalam satu jam, solusinya melewatkan pelatihan otomatis pertama. Untuk informasi selengkapnya, lihat Mengonfigurasi pelatihan otomatis.

Untuk mematikan pelatihan otomatis, atur `performAutoTraining` ke false. Jika Anda mematikan pelatihan otomatis, Anda harus membuat versi solusi secara manual dengan memanggil CreateSolutionVersionoperasi.

Setelah pelatihan dimulai, Anda bisa mendapatkan Amazon Resource Name (ARN) versi solusi dengan operasi ListSolutionVersionsAPI. Untuk mendapatkan statusnya, gunakan DescribeSolutionVersion.

Setelah pelatihan selesai, Anda dapat mengevaluasi akurasi model dengan menelepon GetSolutionMetrics. Ketika Anda puas dengan versi solusi, Anda menerapkannya menggunakan CreateCampaign. Kampanye ini memberikan rekomendasi kepada klien melalui GetRecommendationsAPI.

> **Note**
>
> Amazon Personalize tidak mendukung konfigurasi optimasi hyperparameter `hpoObjective` for solution saat ini.

Status

Solusi dapat berada di salah satu negara berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Untuk mendapatkan status solusinya, hubungi DescribeSolution. Jika Anda menggunakan pelatihan manual, status harus AKTIF sebelum Anda meneleponCreateSolutionVersion.

Terkait APIs

- UpdateSolution
- ListSolutions
- CreateSolutionVersion
- DescribeSolution
- DeleteSolution

- ListSolutionVersions
- DescribeSolutionVersion

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "eventType": "string",
   "name": "string",
   "performAutoML": boolean,
   "performAutoTraining": boolean,
   "performHPO": boolean,
   "recipeArn": "string",
   "solutionConfig": {
```

```
   "algorithmHyperParameters": {
      "string" : "string"
   },
   "autoMLConfig": {
      "metricName": "string",
      "recipeList": [ "string" ]
   },
   "autoTrainingConfig": {
      "schedulingExpression": "string"
   },
   "eventsConfig": {
      "eventParametersList": [
         {
            "eventType": "string",
            "eventValueThreshold": number,
            "weight": number
         }
      ]
   },
   "eventValueThreshold": "string",
   "featureTransformationParameters": {
      "string" : "string"
   },
   "hpoConfig": {
      "algorithmHyperParameterRanges": {
         "categoricalHyperParameterRanges": [
            {
               "name": "string",
               "values": [ "string" ]
            }
         ],
         "continuousHyperParameterRanges": [
            {
               "maxValue": number,
               "minValue": number,
               "name": "string"
            }
         ],
         "integerHyperParameterRanges": [
            {
               "maxValue": number,
               "minValue": number,
               "name": "string"
            }
```

```
            ]
         },
         "hpoObjective": {
            "metricName": "string",
            "metricRegex": "string",
            "type": "string"
         },
         "hpoResourceConfig": {
            "maxNumberOfTrainingJobs": "string",
            "maxParallelTrainingJobs": "string"
         }
      },
      "optimizationObjective": {
         "itemAttribute": "string",
         "objectiveSensitivity": "string"
      },
      "trainingDataConfig": {
         "excludedDatasetColumns": {
            "string" : [ "string" ]
         }
      }
   },
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data yang menyediakan data pelatihan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

eventType

Ketika Anda memiliki beberapa jenis acara (menggunakan bidang `EVENT_TYPE` skema), parameter ini menentukan jenis acara (misalnya, 'klik' atau 'suka') yang digunakan untuk melatih model.

Jika Anda tidak memberikan`eventType`, Amazon Personalize akan menggunakan semua interaksi untuk pelatihan dengan bobot yang sama terlepas dari jenisnya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

name

Nama untuk solusinya.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Ya

performAutoML

> **Important**
>
> Kami tidak menyarankan untuk mengaktifkan pembelajaran mesin otomatis. Sebagai gantinya, cocokkan kasus penggunaan Anda dengan resep Amazon Personalize yang tersedia. Untuk informasi selengkapnya, lihat Memilih resep.

Apakah akan melakukan pembelajaran mesin otomatis (AutoML). Nilai default-nya `false`. Untuk kasus ini, Anda harus menentukan`recipeArn`.

Saat disetel ke`true`, Amazon Personalize menganalisis data latihan Anda dan memilih resep USER_PERSONALIZATION dan hyperparameters yang optimal. Dalam hal ini, Anda harus menghilangkannya`recipeArn`. Amazon Personalize menentukan resep optimal dengan

menjalankan pengujian dengan nilai berbeda untuk hyperparameters. AutoML memperpanjang proses pelatihan dibandingkan dengan memilih resep tertentu.

Tipe: Boolean

Wajib: Tidak

performAutoTraining

Apakah solusi menggunakan pelatihan otomatis untuk membuat versi solusi baru (model terlatih). Defaultnya adalah `True` dan solusinya secara otomatis membuat versi solusi baru setiap 7 hari. Anda dapat mengubah frekuensi pelatihan dengan menentukan a `schedulingExpression` dalam `AutoTrainingConfig` sebagai bagian dari konfigurasi solusi. Untuk informasi selengkapnya tentang pelatihan otomatis, lihat Mengonfigurasi pelatihan otomatis.

Pembuatan versi solusi otomatis dimulai dalam waktu satu jam setelah solusi AKTIF. Jika Anda membuat versi solusi secara manual dalam satu jam, solusinya melewatkan pelatihan otomatis pertama.

Setelah pelatihan dimulai, Anda bisa mendapatkan Amazon Resource Name (ARN) versi solusi dengan operasi ListSolutionVersionsAPI. Untuk mendapatkan statusnya, gunakan DescribeSolutionVersion.

Tipe: Boolean

Wajib: Tidak

performHPO

Apakah akan melakukan optimasi hyperparameter (HPO) pada resep yang ditentukan atau dipilih. Nilai default-nya `false`.

Saat melakukan AutoML, parameter ini selalu `true` dan Anda tidak harus mengaturnya. `false`

Tipe: Boolean

Wajib: Tidak

recipeArn

Nama Sumber Daya Amazon (ARN) dari resep yang akan digunakan untuk pelatihan model. Ini diperlukan ketika `performAutoML` salah. Untuk informasi tentang berbagai resep Amazon Personalisasi dan resep lainnya ARNs, lihat Memilih resep.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

solutionConfig

Properti konfigurasi untuk solusi. Bila `performAutoML` disetel ke true, Amazon Personalize hanya mengevaluasi `autoMLConfig` bagian konfigurasi solusi.

> **Note**
>
> Amazon Personalize tidak mendukung konfigurasi `hpoObjective` saat ini.

Tipe: Objek SolutionConfig

Wajib: Tidak

tags

Daftar tag untuk diterapkan pada solusi.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

Sintaksis Respons

```
{
   "solutionArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

solutionArn

ARN dari solusinya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

CreateSolutionVersion

Layanan: Amazon Personalize

Melatih atau melatih ulang solusi aktif dalam grup kumpulan data Kustom. Solusi dibuat menggunakan CreateSolutionoperasi dan harus dalam keadaan AKTIF sebelum meneleponCreateSolutionVersion. Versi baru dari solusi dibuat setiap kali Anda memanggil operasi ini.

Status

Versi solusi dapat berada di salah satu status berikut:

- BUAT TERTUNDA
- BUAT IN_PROGRESS
- AKTIF
- BUAT GAGAL
- BUAT BERHENTI
- BUAT BERHENTI

Untuk mendapatkan status versi, hubungi DescribeSolutionVersion. Tunggu hingga status ditampilkan sebagai AKTIF sebelum meneleponCreateCampaign.

Jika status ditampilkan sebagai CREATE FAILED, responsnya menyertakan failureReason kunci, yang menjelaskan mengapa pekerjaan gagal.

Terkait APIs

- ListSolutionVersions
- DescribeSolutionVersion
- ListSolutions
- CreateSolution
- DescribeSolution
- DeleteSolution

Sintaksis Permintaan

```
{
```

```
   "name": "string",
   "solutionArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ],
   "trainingMode": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

name

Nama versi solusinya.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

solutionArn

Nama Sumber Daya Amazon (ARN) dari solusi yang berisi informasi konfigurasi pelatihan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

tags

Daftar tag untuk diterapkan ke versi solusi.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Tidak

trainingMode

Ruang lingkup pelatihan yang akan dilakukan saat membuat versi solusi. Nilai default-nya FULL. Ini menciptakan model yang sama sekali baru berdasarkan keseluruhan data pelatihan dari kumpulan data dalam grup kumpulan data Anda.

Jika Anda menggunakan User-Personalization, Anda dapat menentukan mode pelatihan. UPDATE Ini memperbarui model untuk mempertimbangkan item baru untuk rekomendasi. Ini bukan pelatihan ulang penuh. Anda masih harus menyelesaikan pelatihan ulang penuh setiap minggu. Jika Anda menentukanUPDATE, Amazon Personalize akan menghentikan pembaruan otomatis untuk versi solusi. Untuk melanjutkan pembaruan, buat solusi baru dengan mode pelatihan yang disetel ke FULL dan terapkan dalam kampanye. Untuk informasi selengkapnya tentang pembaruan otomatis, lihat Pembaruan otomatis.

UPDATEOpsi ini hanya dapat digunakan ketika Anda sudah memiliki versi solusi aktif yang dibuat dari solusi input menggunakan FULL opsi dan solusi input dilatih dengan resep Personalisasi Pengguna atau resep HRNNN-ColdStart lama.

Tipe: String

Nilai yang Valid: FULL | UPDATE | AUTOTRAIN

Wajib: Tidak

Sintaksis Respons

```
{
   "solutionVersionArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

solutionVersionArn

ARN dari versi solusi baru.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteCampaign

Layanan: Amazon Personalize

Menghapus kampanye dengan menghapus penerapan solusi. Solusi yang menjadi dasar kampanye tidak dihapus dan dapat digunakan kembali saat diperlukan. Kampanye yang dihapus tidak dapat lagi ditentukan dalam GetRecommendationspermintaan. Untuk informasi tentang membuat kampanye, lihat CreateCampaign.

Sintaksis Permintaan

```
{
   "campaignArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

campaignArn

Nama Sumber Daya Amazon (ARN) dari kampanye yang akan dihapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteDataset

Layanan: Amazon Personalize

Menghapus dataset. Anda tidak dapat menghapus kumpulan data jika terkait `DatasetImportJob` atau `SolutionVersion` berada dalam status CREATE PENDING atau IN PROGRESS. Untuk informasi selengkapnya tentang menghapus kumpulan data, lihat Menghapus kumpulan data.

Sintaksis Permintaan

```
{
   "datasetArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data yang akan dihapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteDatasetGroup

Layanan: Amazon Personalize

Menghapus grup dataset. Sebelum menghapus grup kumpulan data, Anda harus menghapus yang berikut ini:

- Semua pelacak acara terkait.
- Semua solusi terkait.
- Semua kumpulan data dalam grup dataset.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

ARN dari grup dataset yang akan dihapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteEventTracker

Layanan: Amazon Personalize

Menghapus pelacak acara. Tidak menghapus dataset dari grup dataset. Untuk informasi lebih lanjut tentang pelacak acara, lihat CreateEventTracker.

Sintaksis Permintaan

```
{
   "eventTrackerArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

eventTrackerArn

Nama Sumber Daya Amazon (ARN) dari pelacak peristiwa yang akan dihapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteFilter

Layanan: Amazon Personalize

Menghapus filter.

Sintaksis Permintaan

```
{
   "filterArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

filterArn

ARN dari filter yang akan dihapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteMetricAttribution

Layanan: Amazon Personalize

Menghapus atribusi metrik.

Sintaksis Permintaan

```
{
   "metricAttributionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

metricAttributionArn

Nama Sumber Daya Amazon (ARN) atribusi metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteRecommender

Layanan: Amazon Personalize

Menonaktifkan dan menghapus pemberi rekomendasi. Rekomendasi yang dihapus tidak dapat lagi ditentukan dalam GetRecommendationspermintaan.

Sintaksis Permintaan

```
{
   "recommenderArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi untuk dihapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteSchema

Layanan: Amazon Personalize

Menghapus skema. Sebelum menghapus skema, Anda harus menghapus semua kumpulan data yang mereferensikan skema. Untuk informasi selengkapnya tentang skema, lihat CreateSchema.

Sintaksis Permintaan

```
{
   "schemaArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

schemaArn

Nama Sumber Daya Amazon (ARN) dari skema yang akan dihapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DeleteSolution

Layanan: Amazon Personalize

Menghapus semua versi solusi dan `Solution` objek itu sendiri. Sebelum menghapus solusi, Anda harus menghapus semua kampanye berdasarkan solusi. Untuk menentukan kampanye apa yang menggunakan solusi, hubungi ListCampaignsdan sediakan Amazon Resource Name (ARN) solusi. Anda tidak dapat menghapus solusi jika terkait `SolutionVersion` berada dalam status CREATE PENDING atau IN PROGRESS. Untuk informasi lebih lanjut tentang solusi, lihat CreateSolution.

Sintaksis Permintaan

```
{
   "solutionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

solutionArn

ARN dari solusi untuk menghapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeAlgorithm

Layanan: Amazon Personalize

Menjelaskan algoritma yang diberikan.

Sintaksis Permintaan

```
{
   "algorithmArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

algorithmArn

Nama Sumber Daya Amazon (ARN) dari algoritma untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "algorithm": {
      "algorithmArn": "string",
      "algorithmImage": {
         "dockerURI": "string",
         "name": "string"
      },
      "creationDateTime": number,
      "defaultHyperParameterRanges": {
         "categoricalHyperParameterRanges": [
            {
               "isTunable": boolean,
               "name": "string",
```

```
                "values": [ "string" ]
            }
         ],
         "continuousHyperParameterRanges": [
            {
               "isTunable": boolean,
               "maxValue": number,
               "minValue": number,
               "name": "string"
            }
         ],
         "integerHyperParameterRanges": [
            {
               "isTunable": boolean,
               "maxValue": number,
               "minValue": number,
               "name": "string"
            }
         ]
      },
      "defaultHyperParameters": {
         "string" : "string"
      },
      "defaultResourceConfig": {
         "string" : "string"
      },
      "lastUpdatedDateTime": number,
      "name": "string",
      "roleArn": "string",
      "trainingInputMode": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

algorithm

Daftar properti algoritme.

Tipe: Objek Algorithm

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeBatchInferenceJob

Layanan: Amazon Personalize

Mendapatkan properti pekerjaan inferensi batch termasuk nama, Amazon Resource Name (ARN), status, konfigurasi input dan output, dan ARN versi solusi yang digunakan untuk menghasilkan rekomendasi.

Sintaksis Permintaan

```
{
   "batchInferenceJobArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

batchInferenceJobArn

ARN dari pekerjaan inferensi batch untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "batchInferenceJob": {
      "batchInferenceJobArn": "string",
      "batchInferenceJobConfig": {
         "itemExplorationConfig": {
            "string" : "string"
         }
      },
      "batchInferenceJobMode": "string",
      "creationDateTime": number,
      "failureReason": "string",
      "filterArn": "string",
```

```
      "jobInput": {
         "s3DataSource": {
            "kmsKeyArn": "string",
            "path": "string"
         }
      },
      "jobName": "string",
      "jobOutput": {
         "s3DataDestination": {
            "kmsKeyArn": "string",
            "path": "string"
         }
      },
      "lastUpdatedDateTime": number,
      "numResults": number,
      "roleArn": "string",
      "solutionVersionArn": "string",
      "status": "string",
      "themeGenerationConfig": {
         "fieldsForThemeGeneration": {
            "itemName": "string"
         }
      }
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

batchInferenceJob

Informasi tentang pekerjaan inferensi batch yang ditentukan.

Tipe: Objek BatchInferenceJob

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeBatchSegmentJob

Layanan: Amazon Personalize

Mendapatkan properti pekerjaan segmen batch termasuk nama, Amazon Resource Name (ARN), status, konfigurasi input dan output, dan ARN versi solusi yang digunakan untuk menghasilkan segmen.

Sintaksis Permintaan

```
{
   "batchSegmentJobArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

batchSegmentJobArn

ARN dari tugas segmen batch untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "batchSegmentJob": {
      "batchSegmentJobArn": "string",
      "creationDateTime": number,
      "failureReason": "string",
      "filterArn": "string",
      "jobInput": {
         "s3DataSource": {
            "kmsKeyArn": "string",
```

```
            "path": "string"
         }
      },
      "jobName": "string",
      "jobOutput": {
         "s3DataDestination": {
            "kmsKeyArn": "string",
            "path": "string"
         }
      },
      "lastUpdatedDateTime": number,
      "numResults": number,
      "roleArn": "string",
      "solutionVersionArn": "string",
      "status": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

batchSegmentJob

Informasi tentang pekerjaan segmen batch yang ditentukan.

Tipe: Objek BatchSegmentJob

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeCampaign

Layanan: Amazon Personalize

Menjelaskan kampanye yang diberikan, termasuk statusnya.

Kampanye dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Ketika `status` ada`CREATE FAILED`, respon termasuk `failureReason` kunci, yang menjelaskan mengapa.

Untuk informasi selengkapnya tentang kampanye, lihat CreateCampaign.

Sintaksis Permintaan

```
{
   "campaignArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

campaignArn

Amazon Resource Name (ARN) dari kampanye.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "campaign": {
```

```
      "campaignArn": "string",
      "campaignConfig": {
         "enableMetadataWithRecommendations": boolean,
         "itemExplorationConfig": {
            "string" : "string"
         },
         "syncWithLatestSolutionVersion": boolean
      },
      "creationDateTime": number,
      "failureReason": "string",
      "lastUpdatedDateTime": number,
      "latestCampaignUpdate": {
         "campaignConfig": {
            "enableMetadataWithRecommendations": boolean,
            "itemExplorationConfig": {
               "string" : "string"
            },
            "syncWithLatestSolutionVersion": boolean
         },
         "creationDateTime": number,
         "failureReason": "string",
         "lastUpdatedDateTime": number,
         "minProvisionedTPS": number,
         "solutionVersionArn": "string",
         "status": "string"
      },
      "minProvisionedTPS": number,
      "name": "string",
      "solutionVersionArn": "string",
      "status": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

campaign

Sifat-sifat kampanye.

Tipe: Objek Campaign

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeDataDeletionJob

Layanan: Amazon Personalize

Menjelaskan pekerjaan penghapusan data yang dibuat oleh CreateDataDeletionJob, termasuk status pekerjaan.

Sintaksis Permintaan

```
{
   "dataDeletionJobArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

dataDeletionJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan penghapusan data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "dataDeletionJob": {
      "creationDateTime": number,
      "dataDeletionJobArn": "string",
      "datasetGroupArn": "string",
      "dataSource": {
         "dataLocation": "string"
      },
      "failureReason": "string",
      "jobName": "string",
      "lastUpdatedDateTime": number,
      "numDeleted": number,
```

```
        "roleArn": "string",
        "status": "string"
    }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

dataDeletionJob

Informasi tentang pekerjaan penghapusan data, termasuk status.

Status adalah salah satu dari nilai berikut:

- MENUNGGU
- IN_PROGRESS
- DISELESAIKAN
- FAILED

Tipe: Objek DataDeletionJob

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeDataset

Layanan: Amazon Personalize

Menjelaskan kumpulan data yang diberikan. Untuk informasi lebih lanjut tentang kumpulan data, lihat. CreateDataset

Sintaksis Permintaan

```
{
   "datasetArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "dataset": {
      "creationDateTime": number,
      "datasetArn": "string",
      "datasetGroupArn": "string",
      "datasetType": "string",
      "lastUpdatedDateTime": number,
      "latestDatasetUpdate": {
         "creationDateTime": number,
         "failureReason": "string",
         "lastUpdatedDateTime": number,
         "schemaArn": "string",
```

```
            "status": "string"
        },
        "name": "string",
        "schemaArn": "string",
        "status": "string",
        "trackingId": "string"
    }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

dataset

Daftar properti dataset.

Tipe: Objek Dataset

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeDatasetExportJob

Layanan: Amazon Personalize

Menjelaskan pekerjaan ekspor kumpulan data yang dibuat oleh CreateDatasetExportJob, termasuk status pekerjaan ekspor.

Sintaksis Permintaan

```
{
   "datasetExportJobArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetExportJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan ekspor dataset untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "datasetExportJob": {
      "creationDateTime": number,
      "datasetArn": "string",
      "datasetExportJobArn": "string",
      "failureReason": "string",
      "ingestionMode": "string",
      "jobName": "string",
      "jobOutput": {
         "s3DataDestination": {
            "kmsKeyArn": "string",
```

```
            "path": "string"
         }
      },
      "lastUpdatedDateTime": number,
      "roleArn": "string",
      "status": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetExportJob

Informasi tentang pekerjaan ekspor dataset, termasuk status.

Status adalah salah satu dari nilai berikut:

- BUAT TERTUNDA
- BUAT IN_PROGRESS
- AKTIF
- BUAT GAGAL

Tipe: Objek DatasetExportJob

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeDatasetGroup

Layanan: Amazon Personalize

Menjelaskan grup kumpulan data yang diberikan. Untuk informasi selengkapnya tentang grup kumpulan data, lihat CreateDatasetGroup.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "datasetGroup": {
      "creationDateTime": number,
      "datasetGroupArn": "string",
      "domain": "string",
      "failureReason": "string",
      "kmsKeyArn": "string",
      "lastUpdatedDateTime": number,
      "name": "string",
      "roleArn": "string",
      "status": "string"
   }
```

```
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetGroup

Daftar properti grup dataset.

Tipe: Objek DatasetGroup

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin

- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeDatasetImportJob

Layanan: Amazon Personalize

Menjelaskan pekerjaan impor kumpulan data yang dibuat oleh CreateDatasetImportJob, termasuk status pekerjaan impor.

Sintaksis Permintaan

```
{
   "datasetImportJobArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetImportJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan impor dataset untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "datasetImportJob": {
      "creationDateTime": number,
      "datasetArn": "string",
      "datasetImportJobArn": "string",
      "dataSource": {
         "dataLocation": "string"
      },
      "failureReason": "string",
      "importMode": "string",
      "jobName": "string",
      "lastUpdatedDateTime": number,
```

```
      "publishAttributionMetricsToS3": boolean,
      "roleArn": "string",
      "status": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetImportJob

Informasi tentang pekerjaan impor dataset, termasuk status.

Status adalah salah satu dari nilai berikut:

- BUAT TERTUNDA
- BUAT IN_PROGRESS
- AKTIF
- BUAT GAGAL

Tipe: Objek DatasetImportJob

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeEventTracker

Layanan: Amazon Personalize

Menjelaskan pelacak peristiwa. Tanggapan termasuk `trackingId` dan `status` dari pelacak acara. Untuk informasi lebih lanjut tentang pelacak acara, lihat CreateEventTracker.

Sintaksis Permintaan

```
{
   "eventTrackerArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

eventTrackerArn

Nama Sumber Daya Amazon (ARN) dari pelacak acara untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "eventTracker": {
      "accountId": "string",
      "creationDateTime": number,
      "datasetGroupArn": "string",
      "eventTrackerArn": "string",
      "lastUpdatedDateTime": number,
      "name": "string",
      "status": "string",
      "trackingId": "string"
   }
```

```
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

eventTracker

Objek yang menggambarkan pelacak peristiwa.

Tipe: Objek EventTracker

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin

- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeFeatureTransformation

Layanan: Amazon Personalize

Menjelaskan transformasi fitur yang diberikan.

Sintaksis Permintaan

```
{
   "featureTransformationArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

featureTransformationArn

Nama Sumber Daya Amazon (ARN) dari transformasi fitur untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "featureTransformation": {
      "creationDateTime": number,
      "defaultParameters": {
         "string" : "string"
      },
      "featureTransformationArn": "string",
      "lastUpdatedDateTime": number,
      "name": "string",
      "status": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

featureTransformation

Daftar FeatureTransformation properti.

Tipe: Objek FeatureTransformation

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3

- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeFilter

Layanan: Amazon Personalize

Menjelaskan properti filter.

Sintaksis Permintaan

```
{
   "filterArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

filterArn

ARN dari filter untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "filter": {
      "creationDateTime": number,
      "datasetGroupArn": "string",
      "failureReason": "string",
      "filterArn": "string",
      "filterExpression": "string",
      "lastUpdatedDateTime": number,
      "name": "string",
      "status": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

filter

Detail filter.

Tipe: Objek Filter

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3

- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeMetricAttribution

Layanan: Amazon Personalize

Menjelaskan atribusi metrik.

Sintaksis Permintaan

```
{
   "metricAttributionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

metricAttributionArn

Nama Sumber Daya Amazon (ARN) atribusi metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "metricAttribution": {
      "creationDateTime": number,
      "datasetGroupArn": "string",
      "failureReason": "string",
      "lastUpdatedDateTime": number,
      "metricAttributionArn": "string",
      "metricsOutputConfig": {
         "roleArn": "string",
         "s3DataDestination": {
            "kmsKeyArn": "string",
            "path": "string"
         }
```

```
        },
        "name": "string",
        "status": "string"
    }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

metricAttribution

Rincian atribusi metrik.

Tipe: Objek MetricAttribution

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeRecipe

Layanan: Amazon Personalize

Menjelaskan resep.

Resep berisi tiga item:

- Algoritma yang melatih model.
- Hyperparameter yang mengatur pelatihan.
- Informasi transformasi fitur untuk memodifikasi data input sebelum pelatihan.

Amazon Personalize menyediakan satu set resep yang telah ditentukan. Anda menentukan resep saat membuat solusi dengan CreateSolutionAPI. `CreateSolution`melatih model dengan menggunakan algoritma dalam resep yang ditentukan dan dataset pelatihan. Solusinya, ketika diterapkan sebagai kampanye, dapat memberikan rekomendasi menggunakan GetRecommendationsAPI.

Sintaksis Permintaan

```
{
   "recipeArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

recipeArn

Nama Sumber Daya Amazon (ARN) dari resep untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "recipe": {
      "algorithmArn": "string",
      "creationDateTime": number,
      "description": "string",
      "featureTransformationArn": "string",
      "lastUpdatedDateTime": number,
      "name": "string",
      "recipeArn": "string",
      "recipeType": "string",
      "status": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

recipe

Objek yang menggambarkan resep.

Tipe: Objek Recipe

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeRecommender

Layanan: Amazon Personalize

Menjelaskan pemberi rekomendasi yang diberikan, termasuk statusnya.

Rekomendasi dapat berada di salah satu negara berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- STOP PENDING > STOP IN_PROGRESS > TIDAK AKTIF> MULAI TERTUNDA > MULAI IN_PROGRESS > AKTIF
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Ketika `status` ada`CREATE FAILED`, respon termasuk `failureReason` kunci, yang menjelaskan mengapa.

`modelMetrics`Kuncinya adalah nol ketika pemberi rekomendasi sedang dibuat atau dihapus.

Untuk informasi lebih lanjut tentang pemberi rekomendasi, lihat CreateRecommender.

Sintaksis Permintaan

```
{
   "recommenderArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "recommender": {
      "creationDateTime": number,
      "datasetGroupArn": "string",
      "failureReason": "string",
      "lastUpdatedDateTime": number,
      "latestRecommenderUpdate": {
         "creationDateTime": number,
         "failureReason": "string",
         "lastUpdatedDateTime": number,
         "recommenderConfig": {
            "enableMetadataWithRecommendations": boolean,
            "itemExplorationConfig": {
               "string" : "string"
            },
            "minRecommendationRequestsPerSecond": number,
            "trainingDataConfig": {
               "excludedDatasetColumns": {
                  "string" : [ "string" ]
               }
            }
         },
         "status": "string"
      },
      "modelMetrics": {
         "string" : number
      },
      "name": "string",
      "recipeArn": "string",
      "recommenderArn": "string",
      "recommenderConfig": {
         "enableMetadataWithRecommendations": boolean,
         "itemExplorationConfig": {
            "string" : "string"
         },
         "minRecommendationRequestsPerSecond": number,
         "trainingDataConfig": {
            "excludedDatasetColumns": {
               "string" : [ "string" ]
            }
         }
      },
```

```
        "status": "string"
    }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

recommender

Sifat-sifat pemberi rekomendasi.

Tipe: Objek Recommender

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2

- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeSchema

Layanan: Amazon Personalize

Menjelaskan skema. Untuk informasi selengkapnya tentang skema, lihat CreateSchema.

Sintaksis Permintaan

```
{
   "schemaArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

schemaArn

Nama Sumber Daya Amazon (ARN) dari skema yang akan diambil.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "schema": {
      "creationDateTime": number,
      "domain": "string",
      "lastUpdatedDateTime": number,
      "name": "string",
      "schema": "string",
      "schemaArn": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

schema

Skema yang diminta.

Tipe: Objek DatasetSchema

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeSolution

Layanan: Amazon Personalize

Menjelaskan solusi. Untuk informasi lebih lanjut tentang solusi, lihat CreateSolution.

Sintaksis Permintaan

```
{
   "solutionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

solutionArn

Nama Sumber Daya Amazon (ARN) dari solusi untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "solution": {
      "autoMLResult": {
         "bestRecipeArn": "string"
      },
      "creationDateTime": number,
      "datasetGroupArn": "string",
      "eventType": "string",
      "lastUpdatedDateTime": number,
      "latestSolutionUpdate": {
         "creationDateTime": number,
         "failureReason": "string",
         "lastUpdatedDateTime": number,
         "performAutoTraining": boolean,
```

```
      "solutionUpdateConfig": {
         "autoTrainingConfig": {
            "schedulingExpression": "string"
         },
         "eventsConfig": {
            "eventParametersList": [
               {
                  "eventType": "string",
                  "eventValueThreshold": number,
                  "weight": number
               }
            ]
         }
      },
      "status": "string"
   },
   "latestSolutionVersion": {
      "creationDateTime": number,
      "failureReason": "string",
      "lastUpdatedDateTime": number,
      "solutionVersionArn": "string",
      "status": "string",
      "trainingMode": "string",
      "trainingType": "string"
   },
   "name": "string",
   "performAutoML": boolean,
   "performAutoTraining": boolean,
   "performHPO": boolean,
   "recipeArn": "string",
   "solutionArn": "string",
   "solutionConfig": {
      "algorithmHyperParameters": {
         "string" : "string"
      },
      "autoMLConfig": {
         "metricName": "string",
         "recipeList": [ "string" ]
      },
      "autoTrainingConfig": {
         "schedulingExpression": "string"
      },
      "eventsConfig": {
         "eventParametersList": [
```

```
            {
               "eventType": "string",
               "eventValueThreshold": number,
               "weight": number
            }
         ]
      },
      "eventValueThreshold": "string",
      "featureTransformationParameters": {
         "string" : "string"
      },
      "hpoConfig": {
         "algorithmHyperParameterRanges": {
            "categoricalHyperParameterRanges": [
               {
                  "name": "string",
                  "values": [ "string" ]
               }
            ],
            "continuousHyperParameterRanges": [
               {
                  "maxValue": number,
                  "minValue": number,
                  "name": "string"
               }
            ],
            "integerHyperParameterRanges": [
               {
                  "maxValue": number,
                  "minValue": number,
                  "name": "string"
               }
            ]
         },
         "hpoObjective": {
            "metricName": "string",
            "metricRegex": "string",
            "type": "string"
         },
         "hpoResourceConfig": {
            "maxNumberOfTrainingJobs": "string",
            "maxParallelTrainingJobs": "string"
         }
      },
```

```
         "optimizationObjective": {
            "itemAttribute": "string",
            "objectiveSensitivity": "string"
         },
         "trainingDataConfig": {
            "excludedDatasetColumns": {
               "string" : [ "string" ]
            }
         }
      },
      "status": "string"
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

solution

Objek yang menggambarkan solusinya.

Tipe: Objek Solution

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

DescribeSolutionVersion

Layanan: Amazon Personalize

Menjelaskan versi spesifik dari solusi. Untuk informasi lebih lanjut tentang solusi, lihat CreateSolution

Sintaksis Permintaan

```
{
   "solutionVersionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "solutionVersion": {
      "creationDateTime": number,
      "datasetGroupArn": "string",
      "eventType": "string",
      "failureReason": "string",
      "lastUpdatedDateTime": number,
      "name": "string",
      "performAutoML": boolean,
      "performHPO": boolean,
      "recipeArn": "string",
      "solutionArn": "string",
      "solutionConfig": {
         "algorithmHyperParameters": {
```

```
      "string" : "string"
   },
   "autoMLConfig": {
      "metricName": "string",
      "recipeList": [ "string" ]
   },
   "autoTrainingConfig": {
      "schedulingExpression": "string"
   },
   "eventsConfig": {
      "eventParametersList": [
         {
            "eventType": "string",
            "eventValueThreshold": number,
            "weight": number
         }
      ]
   },
   "eventValueThreshold": "string",
   "featureTransformationParameters": {
      "string" : "string"
   },
   "hpoConfig": {
      "algorithmHyperParameterRanges": {
         "categoricalHyperParameterRanges": [
            {
               "name": "string",
               "values": [ "string" ]
            }
         ],
         "continuousHyperParameterRanges": [
            {
               "maxValue": number,
               "minValue": number,
               "name": "string"
            }
         ],
         "integerHyperParameterRanges": [
            {
               "maxValue": number,
               "minValue": number,
               "name": "string"
            }
         ]
```

```
            },
            "hpoObjective": {
               "metricName": "string",
               "metricRegex": "string",
               "type": "string"
            },
            "hpoResourceConfig": {
               "maxNumberOfTrainingJobs": "string",
               "maxParallelTrainingJobs": "string"
            }
         },
         "optimizationObjective": {
            "itemAttribute": "string",
            "objectiveSensitivity": "string"
         },
         "trainingDataConfig": {
            "excludedDatasetColumns": {
               "string" : [ "string" ]
            }
         }
      },
      "solutionVersionArn": "string",
      "status": "string",
      "trainingHours": number,
      "trainingMode": "string",
      "trainingType": "string",
      "tunedHPOParams": {
         "algorithmHyperParameters": {
            "string" : "string"
         }
      }
   }
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

solutionVersion

Versi solusinya.

Tipe: Objek SolutionVersion

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

GetSolutionMetrics

Layanan: Amazon Personalize

Mendapat metrik untuk versi solusi yang ditentukan.

Sintaksis Permintaan

```
{
   "solutionVersionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi untuk mendapatkan metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "metrics": {
      "string" : number
   },
   "solutionVersionArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

metrics

Metrik untuk versi solusi. Untuk informasi selengkapnya, lihat Mengevaluasi versi solusi dengan metrik.

Tipe: Peta string ke ganda

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

solutionVersionArn

Versi solusi yang sama ARN seperti yang ditentukan dalam permintaan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListBatchInferenceJobs

Layanan: Amazon Personalize

Mendapat daftar pekerjaan inferensi batch yang telah dilakukan dari versi solusi.

Sintaksis Permintaan

```
{
   "maxResults": number,
   "nextToken": "string",
   "solutionVersionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

maxResults

Jumlah maksimum hasil pekerjaan inferensi batch yang akan dikembalikan di setiap halaman. Nilai default-nya adalah 100.

Tipe: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token untuk meminta halaman hasil berikutnya.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi tempat pekerjaan inferensi batch dibuat.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "batchInferenceJobs": [
      {
         "batchInferenceJobArn": "string",
         "batchInferenceJobMode": "string",
         "creationDateTime": number,
         "failureReason": "string",
         "jobName": "string",
         "lastUpdatedDateTime": number,
         "solutionVersionArn": "string",
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

batchInferenceJobs

Daftar yang berisi informasi tentang setiap pekerjaan yang dikembalikan.

Tipe: Array objek BatchInferenceJobSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token yang digunakan untuk mengambil halaman hasil berikutnya. Nilainya adalah `null` ketika tidak ada lagi hasil untuk dikembalikan.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListBatchSegmentJobs

Layanan: Amazon Personalize

Mendapat daftar pekerjaan segmen batch yang telah dilakukan dari versi solusi yang Anda tentukan.

Sintaksis Permintaan

```
{
   "maxResults": number,
   "nextToken": "string",
   "solutionVersionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

maxResults

Jumlah maksimum hasil pekerjaan segmen batch yang akan dikembalikan di setiap halaman. Nilai default-nya adalah 100.

Tipe: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token untuk meminta halaman hasil berikutnya.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi yang digunakan segmen batch untuk menghasilkan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "batchSegmentJobs": [
      {
         "batchSegmentJobArn": "string",
         "creationDateTime": number,
         "failureReason": "string",
         "jobName": "string",
         "lastUpdatedDateTime": number,
         "solutionVersionArn": "string",
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

batchSegmentJobs

Daftar yang berisi informasi tentang setiap pekerjaan yang dikembalikan.

Tipe: Array objek BatchSegmentJobSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token yang digunakan untuk mengambil halaman hasil berikutnya. Nilainya adalah `null` ketika tidak ada lagi hasil untuk dikembalikan.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListCampaigns

Layanan: Amazon Personalize

Mengembalikan daftar kampanye yang menggunakan solusi yang diberikan. Jika solusi tidak ditentukan, semua kampanye yang terkait dengan akun akan dicantumkan. Respons menyediakan properti untuk setiap kampanye, termasuk Amazon Resource Name (ARN). Untuk informasi selengkapnya tentang kampanye, lihat CreateCampaign.

Sintaksis Permintaan

```
{
   "maxResults": number,
   "nextToken": "string",
   "solutionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

maxResults

Jumlah maksimum kampanye yang akan dikembalikan.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke ListCampaignsuntuk mendapatkan rangkaian kampanye berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Wajib: Tidak

solutionArn

Nama Sumber Daya Amazon (ARN) dari solusi untuk membuat daftar kampanye. Jika solusi tidak ditentukan, semua kampanye yang terkait dengan akun akan dicantumkan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "campaigns": [
      {
         "campaignArn": "string",
         "creationDateTime": number,
         "failureReason": "string",
         "lastUpdatedDateTime": number,
         "name": "string",
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

campaigns

Daftar kampanye.

Tipe: Array objek CampaignSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token untuk mendapatkan rangkaian kampanye berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListDataDeletionJobs

Layanan: Amazon Personalize

Mengembalikan daftar pekerjaan penghapusan data untuk grup kumpulan data yang diurutkan berdasarkan waktu pembuatan, dengan yang terbaru terlebih dahulu. Jika grup kumpulan data tidak ditentukan, semua pekerjaan penghapusan data yang terkait dengan akun akan dicantumkan. Respons menyediakan properti untuk setiap pekerjaan, termasuk Amazon Resource Name (ARN). Untuk informasi selengkapnya tentang pekerjaan penghapusan data, lihat Menghapus pengguna.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data untuk mencantumkan pekerjaan penghapusan data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum pekerjaan penghapusan data untuk dikembalikan.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListDataDeletionJobs` untuk mendapatkan set pekerjaan berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "dataDeletionJobs": [
      {
         "creationDateTime": number,
         "dataDeletionJobArn": "string",
         "datasetGroupArn": "string",
         "failureReason": "string",
         "jobName": "string",
         "lastUpdatedDateTime": number,
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

dataDeletionJobs

Daftar pekerjaan penghapusan data.

Tipe: Array objek DataDeletionJobSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token untuk mendapatkan set pekerjaan penghapusan data berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListDatasetExportJobs

Layanan: Amazon Personalize

Mengembalikan daftar pekerjaan ekspor dataset yang menggunakan dataset yang diberikan. Ketika kumpulan data tidak ditentukan, semua pekerjaan ekspor kumpulan data yang terkait dengan akun akan dicantumkan. Respons menyediakan properti untuk setiap pekerjaan ekspor kumpulan data, termasuk Amazon Resource Name (ARN). Untuk informasi lebih lanjut tentang pekerjaan ekspor kumpulan data, lihat CreateDatasetExportJob. Untuk informasi lebih lanjut tentang kumpulan data, lihat. CreateDataset

Sintaksis Permintaan

```
{
   "datasetArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data untuk mencantumkan pekerjaan ekspor kumpulan data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum pekerjaan ekspor dataset yang akan dikembalikan.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token yang dikembalikan dari panggilan sebelumnya ke `ListDatasetExportJobs` untuk mendapatkan set pekerjaan ekspor dataset berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "datasetExportJobs": [
      {
         "creationDateTime": number,
         "datasetExportJobArn": "string",
         "failureReason": "string",
         "jobName": "string",
         "lastUpdatedDateTime": number,
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetExportJobs

Daftar pekerjaan ekspor dataset.

Tipe: Array objek DatasetExportJobSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token untuk mendapatkan set pekerjaan ekspor dataset berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python

- AWS SDK for Ruby V3

ListDatasetGroups

Layanan: Amazon Personalize

Mengembalikan daftar kelompok dataset. Respons menyediakan properti untuk setiap grup kumpulan data, termasuk Amazon Resource Name (ARN). Untuk informasi selengkapnya tentang grup kumpulan data, lihat CreateDatasetGroup.

Sintaksis Permintaan

```
{
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

maxResults

Jumlah maksimum grup dataset yang akan dikembalikan.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListDatasetGroups` untuk mendapatkan kumpulan grup kumpulan data berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
```

```
   "datasetGroups": [
      {
         "creationDateTime": number,
         "datasetGroupArn": "string",
         "domain": "string",
         "failureReason": "string",
         "lastUpdatedDateTime": number,
         "name": "string",
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetGroups

Daftar grup dataset Anda.

Tipe: Array objek DatasetGroupSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token untuk mendapatkan kumpulan grup dataset berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListDatasetImportJobs

Layanan: Amazon Personalize

Mengembalikan daftar pekerjaan impor dataset yang menggunakan dataset yang diberikan. Ketika kumpulan data tidak ditentukan, semua pekerjaan impor dataset yang terkait dengan akun akan dicantumkan. Respons menyediakan properti untuk setiap pekerjaan impor dataset, termasuk Amazon Resource Name (ARN). Untuk informasi selengkapnya tentang pekerjaan impor kumpulan data, lihat CreateDatasetImportJob. Untuk informasi lebih lanjut tentang kumpulan data, lihat. CreateDataset

Sintaksis Permintaan

```
{
   "datasetArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data untuk mencantumkan pekerjaan impor kumpulan data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum pekerjaan impor dataset untuk dikembalikan.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token yang dikembalikan dari panggilan sebelumnya ke `ListDatasetImportJobs` untuk mendapatkan set pekerjaan impor dataset berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "datasetImportJobs": [
      {
         "creationDateTime": number,
         "datasetImportJobArn": "string",
         "failureReason": "string",
         "importMode": "string",
         "jobName": "string",
         "lastUpdatedDateTime": number,
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetImportJobs

Daftar pekerjaan impor dataset.

Tipe: Array objek DatasetImportJobSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token untuk mendapatkan set pekerjaan impor dataset berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python

- AWS SDK for Ruby V3

ListDatasets

Layanan: Amazon Personalize

Mengembalikan daftar dataset yang terkandung dalam kelompok dataset yang diberikan. Respons menyediakan properti untuk setiap kumpulan data, termasuk Amazon Resource Name (ARN). Untuk informasi lebih lanjut tentang kumpulan data, lihat. CreateDataset

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data yang berisi kumpulan data yang akan dicantumkan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum kumpulan data yang akan dikembalikan.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token yang dikembalikan dari panggilan sebelumnya ke `ListDatasets` untuk mendapatkan set pekerjaan impor dataset berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "datasets": [
      {
         "creationDateTime": number,
         "datasetArn": "string",
         "datasetType": "string",
         "lastUpdatedDateTime": number,
         "name": "string",
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasets

Susunan objek `Dataset`. Setiap objek menyediakan informasi metadata.

Tipe: Array objek DatasetSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token untuk mendapatkan kumpulan kumpulan data berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListEventTrackers

Layanan: Amazon Personalize

Mengembalikan daftar pelacak acara yang terkait dengan akun. Respons menyediakan properti untuk setiap pelacak peristiwa, termasuk Amazon Resource Name (ARN) dan ID pelacakan. Untuk informasi lebih lanjut tentang pelacak acara, lihat CreateEventTracker.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

ARN dari grup dataset digunakan untuk memfilter respons.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum pelacak acara untuk kembali.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListEventTrackers` untuk mendapatkan set pelacak acara berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "eventTrackers": [
      {
         "creationDateTime": number,
         "eventTrackerArn": "string",
         "lastUpdatedDateTime": number,
         "name": "string",
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

eventTrackers

Daftar pelacak acara.

Tipe: Array objek EventTrackerSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token untuk mendapatkan rangkaian pelacak acara berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListFilters

Layanan: Amazon Personalize

Daftar semua filter yang termasuk dalam grup dataset tertentu.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

ARN dari grup dataset yang berisi filter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum filter untuk kembali.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListFilters` untuk mendapatkan set filter berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "Filters": [
      {
         "creationDateTime": number,
         "datasetGroupArn": "string",
         "failureReason": "string",
         "filterArn": "string",
         "lastUpdatedDateTime": number,
         "name": "string",
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

Filters

Daftar filter yang dikembalikan.

Tipe: Array objek FilterSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Token untuk mendapatkan set filter berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListMetricAttributionMetrics

Layanan: Amazon Personalize

Daftar metrik untuk atribusi metrik.

Sintaksis Permintaan

```
{
   "maxResults": number,
   "metricAttributionArn": "string",
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

maxResults

Jumlah maksimum metrik untuk kembali dalam satu halaman hasil.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

metricAttributionArn

Nama Sumber Daya Amazon (ARN) dari atribusi metrik untuk mengambil atribut.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

nextToken

Tentukan token pagination dari permintaan sebelumnya untuk mengambil halaman hasil berikutnya.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "metrics": [
      {
         "eventType": "string",
         "expression": "string",
         "metricName": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

metrics

Metrik untuk atribusi metrik yang ditentukan.

Tipe: Array objek MetricAttribute

Anggota Array: Jumlah maksimum 10 item.

nextToken

Tentukan token pagination dari `ListMetricAttributionMetricsResponse` permintaan sebelumnya untuk mengambil halaman hasil berikutnya.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListMetricAttributions

Layanan: Amazon Personalize

Daftar atribusi metrik.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Kumpulan data atribusi metrik mengelompokkan Amazon Resource Name (ARN).

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum atribusi metrik yang akan dikembalikan dalam satu halaman hasil.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Tentukan token pagination dari permintaan sebelumnya untuk mengambil halaman hasil berikutnya.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "metricAttributions": [
      {
         "creationDateTime": number,
         "failureReason": "string",
         "lastUpdatedDateTime": number,
         "metricAttributionArn": "string",
         "name": "string",
         "status": "string"
      }
   ],
   "nextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

metricAttributions

Daftar atribusi metrik.

Tipe: Array objek MetricAttributionSummary

Anggota Array: Jumlah maksimum 100 item.

nextToken

Tentukan token pagination dari permintaan sebelumnya untuk mengambil halaman hasil berikutnya.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListRecipes

Layanan: Amazon Personalize

Mengembalikan daftar resep yang tersedia. Respons menyediakan properti untuk setiap resep, termasuk Nama Sumber Daya Amazon (ARN) resep.

Sintaksis Permintaan

```
{
   "domain": "string",
   "maxResults": number,
   "nextToken": "string",
   "recipeProvider": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

domain

Filter mengembalikan resep berdasarkan domain untuk grup kumpulan data Domain. Hanya resep (Kasus penggunaan grup kumpulan data domain) untuk domain ini yang disertakan dalam respons. Jika Anda tidak menentukan domain, semua resep dikembalikan.

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Wajib: Tidak

maxResults

Jumlah maksimum resep untuk dikembalikan.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListRecipes` untuk mendapatkan set resep berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Wajib: Tidak

recipeProvider

Default-nya adalah `SERVICE`.

Tipe: String

Nilai yang Valid: `SERVICE`

Wajib: Tidak

Sintaksis Respons

```
{
   "nextToken": "string",
   "recipes": [
      {
         "creationDateTime": number,
         "domain": "string",
         "lastUpdatedDateTime": number,
         "name": "string",
         "recipeArn": "string",
         "status": "string"
      }
   ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

nextToken

Token untuk mendapatkan set resep berikutnya.

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

recipes

Daftar resep yang tersedia.

Tipe: Array objek RecipeSummary

Anggota Array: Jumlah maksimum 100 item.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin

- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListRecommenders

Layanan: Amazon Personalize

Mengembalikan daftar pemberi rekomendasi dalam kelompok dataset Domain tertentu. Jika grup kumpulan data Domain tidak ditentukan, semua pemberi rekomendasi yang terkait dengan akun akan dicantumkan. Respons menyediakan properti untuk setiap pemberi rekomendasi, termasuk Amazon Resource Name (ARN). Untuk informasi lebih lanjut tentang pemberi rekomendasi, lihat CreateRecommender.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset Domain untuk mencantumkan rekomendasi. Jika grup kumpulan data Domain tidak ditentukan, semua pemberi rekomendasi yang terkait dengan akun akan dicantumkan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum pemberi rekomendasi untuk kembali.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListRecommenders` untuk mendapatkan set pemberi rekomendasi berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "nextToken": "string",
   "recommenders": [
      {
         "creationDateTime": number,
         "datasetGroupArn": "string",
         "lastUpdatedDateTime": number,
         "name": "string",
         "recipeArn": "string",
         "recommenderArn": "string",
         "recommenderConfig": {
            "enableMetadataWithRecommendations": boolean,
            "itemExplorationConfig": {
               "string" : "string"
            },
            "minRecommendationRequestsPerSecond": number,
            "trainingDataConfig": {
               "excludedDatasetColumns": {
                  "string" : [ "string" ]
               }
            }
         },
         "status": "string"
      }
   ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

nextToken

Token untuk mendapatkan set rekomendasi berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

recommenders

Daftar pemberi rekomendasi.

Tipe: Array objek RecommenderSummary

Anggota Array: Jumlah maksimum 100 item.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListSchemas

Layanan: Amazon Personalize

Mengembalikan daftar skema yang terkait dengan akun. Respons menyediakan properti untuk setiap skema, termasuk Amazon Resource Name (ARN). Untuk informasi selengkapnya tentang skema, lihat CreateSchema.

Sintaksis Permintaan

```
{
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

maxResults

Jumlah maksimum skema untuk kembali.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListSchemas` untuk mendapatkan rangkaian skema berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
```

```
   "nextToken": "string",
   "schemas": [
      {
         "creationDateTime": number,
         "domain": "string",
         "lastUpdatedDateTime": number,
         "name": "string",
         "schemaArn": "string"
      }
   ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

nextToken

Token yang digunakan untuk mendapatkan rangkaian skema berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

schemas

Daftar skema.

Tipe: Array objek DatasetSchemaSummary

Anggota Array: Jumlah maksimum 100 item.

Kesalahan

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListSolutions

Layanan: Amazon Personalize

Mengembalikan daftar solusi dalam kelompok dataset tertentu. Jika grup kumpulan data tidak ditentukan, semua solusi yang terkait dengan akun akan dicantumkan. Respons menyediakan properti untuk setiap solusi, termasuk Amazon Resource Name (ARN). Untuk informasi lebih lanjut tentang solusi, lihat CreateSolution.

Sintaksis Permintaan

```
{
   "datasetGroupArn": "string",
   "maxResults": number,
   "nextToken": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

maxResults

Jumlah maksimum solusi untuk kembali.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListSolutions` untuk mendapatkan rangkaian solusi berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "nextToken": "string",
   "solutions": [
      {
         "creationDateTime": number,
         "lastUpdatedDateTime": number,
         "name": "string",
         "recipeArn": "string",
         "solutionArn": "string",
         "status": "string"
      }
   ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

nextToken

Token untuk mendapatkan rangkaian solusi berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

solutions

Daftar solusi saat ini.

Tipe: Array objek SolutionSummary

Anggota Array: Jumlah maksimum 100 item.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListSolutionVersions

Layanan: Amazon Personalize

Mengembalikan daftar versi solusi untuk solusi yang diberikan. Ketika solusi tidak ditentukan, semua versi solusi yang terkait dengan akun terdaftar. Respons menyediakan properti untuk setiap versi solusi, termasuk Amazon Resource Name (ARN).

Sintaksis Permintaan

```
{
   "maxResults": number,
   "nextToken": "string",
   "solutionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

maxResults

Jumlah maksimum versi solusi untuk kembali.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

nextToken

Token dikembalikan dari panggilan sebelumnya ke `ListSolutionVersions` untuk mendapatkan set versi solusi berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

Wajib: Tidak

solutionArn

Nama Sumber Daya Amazon (ARN) dari solusinya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "nextToken": "string",
   "solutionVersions": [
      {
         "creationDateTime": number,
         "failureReason": "string",
         "lastUpdatedDateTime": number,
         "solutionVersionArn": "string",
         "status": "string",
         "trainingMode": "string",
         "trainingType": "string"
      }
   ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

nextToken

Token untuk mendapatkan set versi solusi berikutnya (jika ada).

Tipe: String

Kendala Panjang: Panjang maksimum 1500.

Pola: `\p{ASCII}{0,1500}`

solutionVersions

Daftar versi solusi yang menjelaskan properti versi.

Tipe: Array objek SolutionVersionSummary

Anggota Array: Jumlah maksimum 100 item.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

InvalidNextTokenException

Token tidak valid.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

ListTagsForResource

Layanan: Amazon Personalize

Dapatkan daftar tag yang dilampirkan ke sumber daya.

Sintaksis Permintaan

```
{
   "resourceArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

resourceArn

Nama Sumber Daya Amazon (ARN) sumber daya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

tags

Tag sumber daya.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript

- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

StartRecommender

Layanan: Amazon Personalize

Memulai pemberi rekomendasi yang tidak aktif. Memulai pemberi rekomendasi tidak membuat model baru, tetapi melanjutkan penagihan dan pelatihan ulang otomatis untuk pemberi rekomendasi.

Sintaksis Permintaan

```
{
   "recommenderArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi untuk memulai.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "recommenderArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi yang Anda mulai.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python

- AWS SDK for Ruby V3

StopRecommender

Layanan: Amazon Personalize

Menghentikan pemberi rekomendasi yang AKTIF. Menghentikan pemberi rekomendasi menghentikan penagihan dan pelatihan ulang otomatis untuk pemberi rekomendasi.

Sintaksis Permintaan

```
{
   "recommenderArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi untuk berhenti.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
   "recommenderArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi yang Anda hentikan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python

- AWS SDK for Ruby V3

StopSolutionVersionCreation

Layanan: Amazon Personalize

Berhenti membuat versi solusi yang berada dalam status CREATE_PENDING atau CREATE IN_PROGRESS.

Bergantung pada status versi solusi saat ini, status versi solusi berubah sebagai berikut:

- CREATE_PENDING > CREATE_STOPS

  atau
- CREATE_IN_PROGRESS > CREATE_STOP> CREATE_STOPS

Anda ditagih untuk semua pelatihan yang diselesaikan sampai Anda menghentikan pembuatan versi solusi. Anda tidak dapat melanjutkan pembuatan versi solusi setelah dihentikan.

Sintaksis Permintaan

```
{
   "solutionVersionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi yang ingin Anda hentikan pembuatannya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

TagResource

Layanan: Amazon Personalize

Tambahkan daftar tag ke sumber daya.

Sintaksis Permintaan

```
{
   "resourceArn": "string",
   "tags": [
      {
         "tagKey": "string",
         "tagValue": "string"
      }
   ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

resourceArn

Nama Sumber Daya Amazon (ARN) sumber daya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

tags

Tag untuk diterapkan ke sumber daya. Untuk informasi selengkapnya, lihat Menandai sumber daya Amazon Personalisasi.

Tipe: Array objek Tag

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Wajib: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagsException

Anda telah melampaui jumlah maksimum tag yang dapat Anda terapkan ke sumber daya ini.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

UntagResource

Layanan: Amazon Personalize

Menghapus tag tertentu yang dilampirkan ke sumber daya. Untuk informasi selengkapnya, lihat Menghapus tag dari Amazon Personalisasi sumber daya.

Sintaksis Permintaan

```
{
   "resourceArn": "string",
   "tagKeys": [ "string" ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

resourceArn

Nama Sumber Daya Amazon (ARN) sumber daya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

tagKeys

Kunci tag yang akan dihapus.

Tipe: Array string

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 200 item.

Batasan Panjang: Panjang minimum 1. Panjang maksimum 128.

Pola: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

TooManyTagKeysException

Permintaan berisi lebih banyak kunci tag daripada yang dapat dikaitkan dengan sumber daya (50 kunci tag per sumber daya).

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript

- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

UpdateCampaign

Layanan: Amazon Personalize

Memperbarui kampanye untuk menerapkan versi solusi yang dilatih ulang dengan kampanye yang ada, mengubah kampanye`minProvisionedTPS`, atau mengubah konfigurasi kampanye Anda. Misalnya, Anda dapat menyetel `enableMetadataWithRecommendations` ke true untuk kampanye yang ada.

Untuk memperbarui kampanye agar mulai secara otomatis menggunakan versi solusi terbaru, tentukan hal berikut:

- Untuk `SolutionVersionArn` parameter, tentukan Nama Sumber Daya Amazon (ARN) solusi Anda dalam `SolutionArn/$LATEST` format.
- Di`campaignConfig`, atur `syncWithLatestSolutionVersion` ke`true`.

Untuk memperbarui kampanye, status kampanye harus AKTIF atau CREATE FAILED. Periksa status kampanye menggunakan DescribeCampaignoperasi.

> **Note**
>
> Anda masih bisa mendapatkan rekomendasi dari kampanye saat pembaruan sedang berlangsung. Kampanye akan menggunakan versi solusi sebelumnya dan konfigurasi kampanye untuk menghasilkan rekomendasi hingga status pembaruan kampanye terbaru`Active`.

Untuk informasi selengkapnya tentang memperbarui kampanye, termasuk contoh kode, lihat Memperbarui kampanye. Untuk informasi selengkapnya tentang kampanye, lihat Membuat kampanye.

Sintaksis Permintaan

```
{
   "campaignArn": "string",
   "campaignConfig": {
      "enableMetadataWithRecommendations": boolean,
      "itemExplorationConfig": {
         "string" : "string"
      },
```

```
        "syncWithLatestSolutionVersion": boolean
    },
    "minProvisionedTPS": number,
    "solutionVersionArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

campaignArn

Amazon Resource Name (ARN) dari kampanye.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

campaignConfig

Detail konfigurasi kampanye.

Tipe: Objek CampaignConfig

Wajib: Tidak

minProvisionedTPS

Menentukan transaksi minimum yang disediakan (rekomendasi) per detik yang akan didukung Amazon Personalize. Tinggi `minProvisionedTPS` akan meningkatkan tagihan Anda. Sebaiknya mulai dengan 1 untuk `minProvisionedTPS` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1.

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari model baru untuk digunakan. Untuk menentukan versi solusi terbaru dari solusi Anda, tentukan ARN solusi Anda dalam *SolutionArn/$LATEST* format. Anda harus menggunakan format ini jika Anda mengatur `syncWithLatestSolutionVersion` ke `True` dalam CampaignConfig.

Untuk menerapkan model yang bukan versi solusi terbaru dari solusi Anda, tentukan ARN versi solusi.

Untuk informasi selengkapnya tentang pembaruan kampanye otomatis, lihat Mengaktifkan pembaruan kampanye otomatis.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Sintaksis Respons

```
{
   "campaignArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

campaignArn

Kampanye ARN yang sama seperti yang diberikan dalam permintaan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

UpdateDataset

Layanan: Amazon Personalize

Perbarui kumpulan data untuk mengganti skema dengan yang baru atau yang sudah ada. Untuk informasi selengkapnya, lihat Mengganti skema kumpulan data.

Sintaksis Permintaan

```
{
   "datasetArn": "string",
   "schemaArn": "string"
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data yang ingin Anda perbarui.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

schemaArn

Nama Sumber Daya Amazon (ARN) dari skema baru yang ingin Anda gunakan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
{
```

```
   "datasetArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data yang Anda perbarui.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

UpdateMetricAttribution

Layanan: Amazon Personalize

Memperbarui atribusi metrik.

Sintaksis Permintaan

```
{
   "addMetrics": [
      {
         "eventType": "string",
         "expression": "string",
         "metricName": "string"
      }
   ],
   "metricAttributionArn": "string",
   "metricsOutputConfig": {
      "roleArn": "string",
      "s3DataDestination": {
         "kmsKeyArn": "string",
         "path": "string"
      }
   },
   "removeMetrics": [ "string" ]
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

addMetrics

Tambahkan atribut metrik baru ke atribusi metrik.

Tipe: Array objek MetricAttribute

Anggota Array: Jumlah maksimum 10 item.

Wajib: Tidak

metricAttributionArn

Nama Sumber Daya Amazon (ARN) untuk atribusi metrik yang akan diperbarui.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

metricsOutputConfig

Konfigurasi keluaran untuk atribusi metrik.

Tipe: Objek MetricAttributionOutput

Wajib: Tidak

removeMetrics

Hapus atribut metrik dari atribusi metrik.

Tipe: Array string

Anggota Array: Jumlah maksimum 10 item.

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Sintaksis Respons

```
{
   "metricAttributionArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

metricAttributionArn

Nama Sumber Daya Amazon (ARN) untuk atribusi metrik yang Anda perbarui.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceAlreadyExistsException

Sumber daya yang ditentukan sudah ada.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2

- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

UpdateRecommender

Layanan: Amazon Personalize

Memperbarui pemberi rekomendasi untuk memodifikasi konfigurasi pemberi rekomendasi. Jika Anda memperbarui pemberi rekomendasi untuk memodifikasi kolom yang digunakan dalam pelatihan, Amazon Personalize secara otomatis memulai pelatihan ulang penuh model yang mendukung pemberi rekomendasi Anda. Sementara pembaruan selesai, Anda masih bisa mendapatkan rekomendasi dari pemberi rekomendasi. Rekomendasi menggunakan konfigurasi sebelumnya hingga pembaruan selesai. Untuk melacak status pembaruan ini, gunakan yang `latestRecommenderUpdate` dikembalikan dalam DescribeRecommenderoperasi.

Sintaksis Permintaan

```
{
   "recommenderArn": "string",
   "recommenderConfig": {
      "enableMetadataWithRecommendations": boolean,
      "itemExplorationConfig": {
         "string" : "string"
      },
      "minRecommendationRequestsPerSecond": number,
      "trainingDataConfig": {
         "excludedDatasetColumns": {
            "string" : [ "string" ]
         }
      }
   }
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi untuk dimodifikasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

recommenderConfig

Detail konfigurasi dari pemberi rekomendasi.

Tipe: Objek RecommenderConfig

Wajib: Ya

Sintaksis Respons

```
{
   "recommenderArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

recommenderArn

Rekomendasi yang sama Amazon Resource Name (ARN) seperti yang diberikan dalam permintaan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

UpdateSolution

Layanan: Amazon Personalize

Memperbarui solusi Amazon Personalize untuk menggunakan konfigurasi pelatihan otomatis yang berbeda. Saat Anda memperbarui solusi, Anda dapat mengubah apakah solusinya menggunakan pelatihan otomatis, dan Anda dapat mengubah frekuensi pelatihan. Untuk informasi selengkapnya tentang memperbarui solusi, lihat Memperbarui solusi.

Pembaruan solusi dapat berada di salah satu status berikut:

BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Untuk mendapatkan status pembaruan solusi, panggil operasi DescribeSolutionAPI dan temukan statusnya di`latestSolutionUpdate`.

Sintaksis Permintaan

```
{
   "performAutoTraining": boolean,
   "solutionArn": "string",
   "solutionUpdateConfig": {
      "autoTrainingConfig": {
         "schedulingExpression": "string"
      },
      "eventsConfig": {
         "eventParametersList": [
            {
               "eventType": "string",
               "eventValueThreshold": number,
               "weight": number
            }
         ]
      }
   }
}
```

Parameter Permintaan

Permintaan menerima data berikut dalam format JSON.

performAutoTraining

Apakah solusi menggunakan pelatihan otomatis untuk membuat versi solusi baru (model terlatih). Anda dapat mengubah frekuensi pelatihan dengan menentukan a `schedulingExpression` dalam `AutoTrainingConfig` sebagai bagian dari konfigurasi solusi.

Jika Anda mengaktifkan pelatihan otomatis, pelatihan otomatis pertama dimulai dalam waktu satu jam setelah pembaruan solusi selesai. Jika Anda membuat versi solusi secara manual dalam satu jam, solusinya melewatkan pelatihan otomatis pertama. Untuk informasi selengkapnya tentang pelatihan otomatis, lihat Mengonfigurasi pelatihan otomatis.

Setelah pelatihan dimulai, Anda bisa mendapatkan Amazon Resource Name (ARN) versi solusi dengan operasi ListSolutionVersionsAPI. Untuk mendapatkan statusnya, gunakan DescribeSolutionVersion.

Tipe: Boolean

Wajib: Tidak

solutionArn

Nama Sumber Daya Amazon (ARN) dari solusi untuk memperbarui.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

solutionUpdateConfig

Detail konfigurasi baru dari solusi.

Tipe: Objek SolutionUpdateConfig

Wajib: Tidak

Sintaksis Respons

```
{
```

```
    "solutionArn": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

solutionArn

Solusi yang sama Amazon Resource Name (ARN) seperti yang diberikan dalam permintaan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

LimitExceededException

Batas jumlah permintaan per detik telah terlampaui.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 400

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

Kejadian Amazon Personalize

Tindakan berikut didukung oleh Amazon Personalize Events:

- PutActionInteractions
- PutActions
- PutEvents
- PutItems
- PutUsers

PutActionInteractions

Layanan: Amazon Personalize Events

Merekam data peristiwa interaksi tindakan. Peristiwa interaksi tindakan adalah interaksi antara pengguna dan tindakan. Misalnya, pengguna mengambil tindakan, seperti mendaftar dalam program keanggotaan atau mengunduh aplikasi Anda.

Untuk informasi selengkapnya tentang merekam interaksi tindakan, lihat Merekam peristiwa interaksi tindakan. Untuk informasi selengkapnya tentang tindakan dalam kumpulan data Tindakan, lihat Kumpulan data Tindakan.

Minta Sintaks

```
POST /action-interactions HTTP/1.1
Content-type: application/json

{
   "actionInteractions": [
      {
         "actionId": "string",
         "eventId": "string",
         "eventType": "string",
         "impression": [ "string" ],
         "properties": "string",
         "recommendationId": "string",
         "sessionId": "string",
         "timestamp": number,
         "userId": "string"
      }
   ],
   "trackingId": "string"
}
```

Parameter Permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi Permintaan

Permintaan menerima data berikut dalam format JSON.

actionInteractions

Daftar peristiwa interaksi tindakan dari sesi.

Tipe: Array objek ActionInteraction

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Wajib: Ya

trackingId

ID pelacak peristiwa interaksi tindakan Anda. Saat Anda membuat kumpulan data interaksi Tindakan, Amazon Personalize akan membuat pelacak peristiwa interaksi tindakan untuk Anda. Untuk informasi selengkapnya, lihat ID pelacak peristiwa interaksi tindakan.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

Sintaksis Respons

```
HTTP/1.1 200
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 409

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 404

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

PutActions

Layanan: Amazon Personalize Events

Menambahkan satu atau beberapa tindakan ke kumpulan data Tindakan. Untuk informasi selengkapnya, lihat Mengimpor tindakan secara individual.

Minta Sintaks

```
POST /actions HTTP/1.1
Content-type: application/json

{
   "actions": [
      {
         "actionId": "string",
         "properties": "string"
      }
   ],
   "datasetArn": "string"
}
```

Parameter Permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi Permintaan

Permintaan menerima data berikut dalam format JSON.

actions

Daftar data tindakan.

Tipe: Array objek Action

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Wajib: Ya

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data Tindakan yang Anda tambahkan tindakan atau tindakan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Ya

Sintaksis Respons

```
HTTP/1.1 200
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 409

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 404

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

PutEvents

Layanan: Amazon Personalize Events

Merekam data peristiwa interaksi item. Untuk informasi selengkapnya lihat Merekam peristiwa interaksi item.

> **Note**
>
> Jika Anda menggunakan AWS Lambda fungsi untuk memanggil PutEvents operasi, peran eksekusi fungsi Anda harus memiliki izin untuk melakukan `personalize:PutEvents` tindakan dengan wildcard `*` dalam `Resource` elemen.

Minta Sintaks

```
POST /events HTTP/1.1
Content-type: application/json

{
   "eventList": [
      {
         "eventId": "string",
         "eventType": "string",
         "eventValue": number,
         "impression": [ "string" ],
         "itemId": "string",
         "metricAttribution": {
            "eventAttributionSource": "string"
         },
         "properties": "string",
         "recommendationId": "string",
         "sentAt": number
      }
   ],
   "sessionId": "string",
   "trackingId": "string",
   "userId": "string"
}
```

Parameter Permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi Permintaan

Permintaan menerima data berikut dalam format JSON.

eventList

Daftar data acara dari sesi.

Tipe: Array objek Event

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Wajib: Ya

sessionId

ID sesi yang terkait dengan kunjungan pengguna. Aplikasi Anda menghasilkan SessionID saat pengguna pertama kali mengunjungi situs web Anda atau menggunakan aplikasi Anda. Amazon Personalize menggunakan SessionID untuk mengaitkan peristiwa dengan pengguna sebelum masuk. Untuk informasi selengkapnya, lihat Merekam peristiwa interaksi item.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

trackingId

ID pelacakan untuk acara tersebut. ID dihasilkan oleh panggilan ke CreateEventTrackerAPI.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

userId

Pengguna yang terkait dengan acara tersebut.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Tidak

Sintaksis Respons

```
HTTP/1.1 200
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

PutItems

Layanan: Amazon Personalize Events

Menambahkan satu atau beberapa item ke dataset Item. Untuk informasi selengkapnya lihat Mengimpor item satu per satu.

Minta Sintaks

```
POST /items HTTP/1.1
Content-type: application/json

{
   "datasetArn": "string",
   "items": [
      {
         "itemId": "string",
         "properties": "string"
      }
   ]
}
```

Parameter Permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data Item tempat Anda menambahkan item atau item.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

items

Daftar data item.

Tipe: Array objek Item

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Wajib: Ya

Sintaksis Respons

```
HTTP/1.1 200
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 409

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 404

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

PutUsers

Layanan: Amazon Personalize Events

Menambahkan satu atau beberapa pengguna ke kumpulan data Pengguna. Untuk informasi selengkapnya lihat Mengimpor pengguna satu per satu.

Minta Sintaks

```
POST /users HTTP/1.1
Content-type: application/json

{
   "datasetArn": "string",
   "users": [
      {
         "properties": "string",
         "userId": "string"
      }
   ]
}
```

Parameter Permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi Permintaan

Permintaan menerima data berikut dalam format JSON.

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data Pengguna yang Anda tambahkan pengguna atau pengguna.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

users

Daftar data pengguna.

Tipe: Array objek User

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Wajib: Ya

Sintaksis Respons

```
HTTP/1.1 200
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan isi HTTP kosong.

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceInUseException

Sumber daya yang ditentukan sedang digunakan.

Kode Status HTTP: 409

ResourceNotFoundException

Tidak dapat menemukan sumber daya yang ditentukan.

Kode Status HTTP: 404

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

Waktu Aktif Amazon Personalize

Tindakan berikut didukung oleh Amazon Personalize Runtime:

- GetActionRecommendations
- GetPersonalizedRanking
- GetRecommendations

GetActionRecommendations

Layanan: Amazon Personalize Runtime

Mengembalikan daftar tindakan yang direkomendasikan diurutkan dalam urutan menurun berdasarkan skor prediksi. Gunakan `GetActionRecommendations` API jika Anda memiliki kampanye khusus yang menerapkan versi solusi yang dilatih dengan resep PERSONALIZED_ACTIONS.

Untuk informasi selengkapnya tentang resep PERSONALIZED_ACTIONS, lihat resep PERSONALIZED_ACTIONS. Untuk informasi selengkapnya tentang mendapatkan rekomendasi tindakan, lihat Mendapatkan rekomendasi tindakan.

Minta Sintaks

```
POST /action-recommendations HTTP/1.1
Content-type: application/json

{
   "campaignArn": "string",
   "filterArn": "string",
   "filterValues": {
      "string" : "string"
   },
   "numResults": number,
   "userId": "string"
}
```

Parameter Permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi Permintaan

Permintaan menerima data berikut dalam format JSON.

campaignArn

Nama Sumber Daya Amazon (ARN) kampanye yang akan digunakan untuk mendapatkan rekomendasi tindakan. Kampanye ini harus menerapkan versi solusi yang dilatih dengan resep PERSONALIZED_ACTIONS.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

filterArn

ARN filter untuk diterapkan pada rekomendasi yang dikembalikan. Untuk informasi selengkapnya, lihat Memfilter Rekomendasi.

Saat menggunakan parameter ini, pastikan sumber daya filternya`ACTIVE`.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

filterValues

Nilai yang digunakan saat memfilter rekomendasi. Untuk setiap parameter placeholder dalam ekspresi filter Anda, berikan nama parameter (dalam kasus yang cocok) sebagai kunci dan nilai filter sebagai nilai yang sesuai. Pisahkan beberapa nilai untuk satu parameter dengan koma.

Untuk ekspresi filter yang menggunakan `INCLUDE` elemen untuk menyertakan tindakan, Anda harus memberikan nilai untuk semua parameter yang ditentukan dalam ekspresi. Untuk filter dengan ekspresi yang menggunakan `EXCLUDE` elemen untuk mengecualikan tindakan, Anda dapat menghilangkan. `filter-values` Dalam kasus ini, Amazon Personalize tidak menggunakan bagian ekspresi tersebut untuk memfilter rekomendasi.

Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 25 item.

Kendala Panjang Kunci: Panjang maksimum 50.

Pola Kunci: `[A-Za-z0-9_]+`

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

numResults

Jumlah hasil yang akan dikembalikan. Default-nya adalah 5. Maksimal adalah 100.

Jenis: Integer

Rentang yang Valid: Nilai minimum 0.

Wajib: Tidak

userId

ID pengguna pengguna untuk memberikan rekomendasi tindakan untuk.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Sintaksis Respons

```
HTTP/1.1 200
Content-type: application/json

{
   "actionList": [
      {
         "actionId": "string",
         "score": number
      }
   ],
   "recommendationId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

actionList

Daftar rekomendasi tindakan yang diurutkan dalam urutan menurun berdasarkan skor prediksi. Bisa ada maksimal 100 tindakan dalam daftar. Untuk informasi tentang skor aksi, lihat Cara kerja penilaian rekomendasi tindakan.

Tipe: Array objek PredictedAction

recommendationId

ID rekomendasi.

Tipe: String

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Sumber daya yang ditentukan tidak ada.

Kode Status HTTP: 404

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin

- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

GetPersonalizedRanking

Layanan: Amazon Personalize Runtime

Memberi peringkat ulang daftar item yang direkomendasikan untuk pengguna tertentu. Item pertama dalam daftar dianggap item yang paling mungkin menarik bagi pengguna.

> **Note**
>
> Solusi yang mendukung kampanye harus dibuat menggunakan resep tipe PERSONALIZED_RANKING.

Minta Sintaks

```
POST /personalize-ranking HTTP/1.1
Content-type: application/json

{
   "campaignArn": "string",
   "context": {
      "string" : "string"
   },
   "filterArn": "string",
   "filterValues": {
      "string" : "string"
   },
   "inputList": [ "string" ],
   "metadataColumns": {
      "string" : [ "string" ]
   },
   "userId": "string"
}
```

Parameter Permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi Permintaan

Permintaan menerima data berikut dalam format JSON.

campaignArn

Nama Sumber Daya Amazon (ARN) kampanye yang akan digunakan untuk menghasilkan peringkat yang dipersonalisasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Ya

context

Metadata kontekstual yang digunakan saat mendapatkan rekomendasi. Metadata kontekstual mencakup informasi interaksi apa pun yang mungkin relevan saat mendapatkan rekomendasi pengguna, seperti lokasi pengguna saat ini atau jenis perangkat.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 150 item.

Kendala Panjang Kunci: Panjang maksimum 150.

Pola Kunci: `[A-Za-z\d_]+`

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

filterArn

Nama Sumber Daya Amazon (ARN) dari filter yang Anda buat untuk menyertakan item atau mengecualikan item dari rekomendasi untuk pengguna tertentu. Untuk informasi selengkapnya, lihat Memfilter Rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

filterValues

Nilai yang digunakan saat memfilter rekomendasi. Untuk setiap parameter placeholder dalam ekspresi filter Anda, berikan nama parameter (dalam kasus yang cocok) sebagai kunci dan nilai filter sebagai nilai yang sesuai. Pisahkan beberapa nilai untuk satu parameter dengan koma.

Untuk ekspresi filter yang menggunakan `INCLUDE` elemen untuk menyertakan item, Anda harus memberikan nilai untuk semua parameter yang ditentukan dalam ekspresi. Untuk filter dengan ekspresi yang menggunakan `EXCLUDE` elemen untuk mengecualikan item, Anda dapat menghilangkan `filter-values` .Dalam kasus ini, Amazon Personalize tidak menggunakan bagian ekspresi tersebut untuk memfilter rekomendasi.

Untuk informasi selengkapnya, lihat Memfilter Rekomendasi.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 25 item.

Kendala Panjang Kunci: Panjang maksimum 50.

Pola Kunci: `[A-Za-z0-9_]+`

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

inputList

Daftar item (oleh`itemId`) ke peringkat. Jika item tidak termasuk dalam kumpulan data pelatihan, item tersebut ditambahkan ke akhir daftar yang di-reanked. Jika Anda memasukkan metadata dalam rekomendasi, maksimumnya adalah 50. Kalau tidak, maksimumnya adalah 500.

Tipe: Array string

Batasan Panjang: Panjang maksimum 256.

Wajib: Ya

metadataColumns

Jika Anda mengaktifkan metadata dalam rekomendasi saat membuat atau memperbarui kampanye, tentukan kolom metadata dari kumpulan data Item untuk disertakan dalam peringkat yang dipersonalisasi. Kunci peta adalah `ITEMS` dan nilainya adalah daftar nama kolom dari kumpulan data Item Anda. Jumlah maksimum kolom yang dapat Anda berikan adalah 10.

Untuk informasi tentang mengaktifkan metadata untuk kampanye, lihat Mengaktifkan metadata dalam rekomendasi untuk kampanye.

Tipe: Peta string ke array string

Entri Peta: Jumlah maksimum 1 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Anggota Array: Jumlah maksimum 99 item.

Kendala Panjang: Panjang maksimum 150.

Wajib: Tidak

userId

Pengguna yang Anda inginkan kampanyenya memberikan peringkat yang dipersonalisasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Ya

Sintaksis Respons

```
HTTP/1.1 200
Content-type: application/json

{
   "personalizedRanking": [
      {
         "itemId": "string",
         "metadata": {
            "string" : "string"
         },
         "promotionName": "string",
         "reason": [ "string" ],
         "score": number
      }
   ],
   "recommendationId": "string"
```

```
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

personalizedRanking

Daftar item dalam urutan yang paling mungkin menarik bagi pengguna. Maksimal adalah 500.

Tipe: Array objek PredictedItem

recommendationId

ID rekomendasi.

Tipe: String

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Sumber daya yang ditentukan tidak ada.

Kode Status HTTP: 404

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

GetRecommendations

Layanan: Amazon Personalize Runtime

Mengembalikan daftar item yang direkomendasikan. Untuk kampanye, Nama Sumber Daya Amazon (ARN) kampanye diperlukan dan input pengguna dan item yang diperlukan bergantung pada jenis resep yang digunakan untuk membuat solusi yang mendukung kampanye sebagai berikut:

- USER_PERSONALIZATION - `userId` diperlukan, tidak digunakan `itemId`
- RELATED_ITEMS - `itemId` wajib, tidak digunakan `userId`

> **Note**
>
> Kampanye yang didukung oleh solusi yang dibuat menggunakan resep tipe PERSONALIZED_RANKING menggunakan API. GetPersonalizedRanking

Untuk pemberi rekomendasi, ARN pemberi rekomendasi diperlukan dan item yang diperlukan serta input pengguna tergantung pada kasus penggunaan (resep berbasis domain) yang mendukung pemberi rekomendasi. Untuk informasi tentang persyaratan kasus penggunaan, lihat Memilih kasus penggunaan pemberi rekomendasi.

Minta Sintaks

```
POST /recommendations HTTP/1.1
Content-type: application/json

{
   "campaignArn": "string",
   "context": {
      "string" : "string"
   },
   "filterArn": "string",
   "filterValues": {
      "string" : "string"
   },
   "itemId": "string",
   "metadataColumns": {
      "string" : [ "string" ]
   },
   "numResults": number,
```

```
   "promotions": [
      {
         "filterArn": "string",
         "filterValues": {
            "string" : "string"
         },
         "name": "string",
         "percentPromotedItems": number
      }
   ],
   "recommenderArn": "string",
   "userId": "string"
}
```

Parameter Permintaan URI

Permintaan tidak menggunakan parameter URI apa pun.

Isi Permintaan

Permintaan menerima data berikut dalam format JSON.

campaignArn

Nama Sumber Daya Amazon (ARN) dari kampanye yang akan digunakan untuk mendapatkan rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

context

Metadata kontekstual yang digunakan saat mendapatkan rekomendasi. Metadata kontekstual mencakup informasi interaksi apa pun yang mungkin relevan saat mendapatkan rekomendasi pengguna, seperti lokasi pengguna saat ini atau jenis perangkat.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 150 item.

Kendala Panjang Kunci: Panjang maksimum 150.

Pola Kunci: `[A-Za-z\d_]+`

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

filterArn

ARN filter untuk diterapkan pada rekomendasi yang dikembalikan. Untuk informasi selengkapnya, lihat Memfilter Rekomendasi.

Saat menggunakan parameter ini, pastikan sumber daya filternya`ACTIVE`.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

filterValues

Nilai yang digunakan saat memfilter rekomendasi. Untuk setiap parameter placeholder dalam ekspresi filter Anda, berikan nama parameter (dalam kasus yang cocok) sebagai kunci dan nilai filter sebagai nilai yang sesuai. Pisahkan beberapa nilai untuk satu parameter dengan koma.

Untuk ekspresi filter yang menggunakan `INCLUDE` elemen untuk menyertakan item, Anda harus memberikan nilai untuk semua parameter yang ditentukan dalam ekspresi. Untuk filter dengan ekspresi yang menggunakan `EXCLUDE` elemen untuk mengecualikan item, Anda dapat menghilangkan `filter-values` .Dalam kasus ini, Amazon Personalize tidak menggunakan bagian ekspresi tersebut untuk memfilter rekomendasi.

Untuk informasi selengkapnya, lihat Memfilter rekomendasi dan segmen pengguna.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 25 item.

Kendala Panjang Kunci: Panjang maksimum 50.

Pola Kunci: `[A-Za-z0-9_]+`

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

itemId

ID item untuk memberikan rekomendasi untuk.

Diperlukan untuk jenis `RELATED_ITEMS` resep.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

metadataColumns

Jika Anda mengaktifkan metadata dalam rekomendasi saat membuat atau memperbarui kampanye atau pemberi rekomendasi, tentukan kolom metadata dari kumpulan data Item untuk disertakan dalam rekomendasi item. Kunci peta adalah `ITEMS` dan nilainya adalah daftar nama kolom dari kumpulan data Item Anda. Jumlah maksimum kolom yang dapat Anda berikan adalah 10.

Untuk informasi tentang mengaktifkan metadata untuk kampanye, lihat Mengaktifkan metadata dalam rekomendasi untuk kampanye. Untuk informasi tentang mengaktifkan metadata untuk pemberi rekomendasi, lihat Mengaktifkan metadata dalam rekomendasi untuk pemberi rekomendasi.

Tipe: Peta string ke array string

Entri Peta: Jumlah maksimum 1 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Anggota Array: Jumlah maksimum 99 item.

Kendala Panjang: Panjang maksimum 150.

Wajib: Tidak

numResults

Jumlah hasil yang akan dikembalikan. Default-nya adalah 25. Jika Anda memasukkan metadata dalam rekomendasi, maksimumnya adalah 50. Kalau tidak, maksimumnya adalah 500.

Jenis: Integer

Rentang yang Valid: Nilai minimum 0.

Wajib: Tidak

promotions

Promosi untuk diterapkan pada permintaan rekomendasi. Promosi mendefinisikan aturan bisnis tambahan yang berlaku untuk subset item yang direkomendasikan yang dapat dikonfigurasi.

Tipe: Array objek Promotion

Anggota Array: Jumlah maksimum 1 item.

Wajib: Tidak

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi untuk digunakan untuk mendapatkan rekomendasi. Berikan ARN pemberi rekomendasi jika Anda membuat grup kumpulan data Domain dengan pemberi rekomendasi untuk kasus penggunaan domain.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

userId

ID pengguna untuk memberikan rekomendasi untuk.

Diperlukan untuk jenis `USER_PERSONALIZATION` resep.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Sintaksis Respons

```
HTTP/1.1 200
```

```
Content-type: application/json

{
   "itemList": [
      {
         "itemId": "string",
         "metadata": {
            "string" : "string"
         },
         "promotionName": "string",
         "reason": [ "string" ],
         "score": number
      }
   ],
   "recommendationId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Layanan mengembalikan data berikut dalam format JSON.

itemList

Daftar rekomendasi yang diurutkan dalam urutan menurun berdasarkan skor prediksi. Bisa ada maksimal 500 item dalam daftar.

Tipe: Array objek PredictedItem

recommendationId

ID rekomendasi.

Tipe: String

Kesalahan

InvalidInputException

Berikan nilai yang valid untuk bidang atau parameter.

Kode Status HTTP: 400

ResourceNotFoundException

Sumber daya yang ditentukan tidak ada.

Kode Status HTTP: 404

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS Antarmuka Baris Perintah
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK untuk V3 JavaScript
- AWS SDK para Kotlin
- AWS SDK for PHP V3
- AWS SDK untuk Python
- AWS SDK for Ruby V3

Tipe Data

Tipe data berikut ini didukung oleh Amazon Personalize:

- Algorithm
- AlgorithmImage
- AutoMLConfig
- AutoMLResult
- AutoTrainingConfig
- BatchInferenceJob
- BatchInferenceJobConfig
- BatchInferenceJobInput

- BatchInferenceJobOutput
- BatchInferenceJobSummary
- BatchSegmentJob
- BatchSegmentJobInput
- BatchSegmentJobOutput
- BatchSegmentJobSummary
- Campaign
- CampaignConfig
- CampaignSummary
- CampaignUpdateSummary
- CategoricalHyperParameterRange
- ContinuousHyperParameterRange
- DataDeletionJob
- DataDeletionJobSummary
- Dataset
- DatasetExportJob
- DatasetExportJobOutput
- DatasetExportJobSummary
- DatasetGroup
- DatasetGroupSummary
- DatasetImportJob
- DatasetImportJobSummary
- DatasetSchema
- DatasetSchemaSummary
- DatasetSummary
- DatasetUpdateSummary
- DataSource
- DefaultCategoricalHyperParameterRange
- DefaultContinuousHyperParameterRange
- DefaultHyperParameterRanges

- DefaultIntegerHyperParameterRange
- EventParameters
- EventsConfig
- EventTracker
- EventTrackerSummary
- FeatureTransformation
- FieldsForThemeGeneration
- Filter
- FilterSummary
- HPOConfig
- HPOObjective
- HPOResourceConfig
- HyperParameterRanges
- IntegerHyperParameterRange
- MetricAttribute
- MetricAttribution
- MetricAttributionOutput
- MetricAttributionSummary
- OptimizationObjective
- Recipe
- RecipeSummary
- Recommender
- RecommenderConfig
- RecommenderSummary
- RecommenderUpdateSummary
- S3DataConfig
- Solution
- SolutionConfig
- SolutionSummary
- SolutionUpdateConfig

- SolutionUpdateSummary
- SolutionVersion
- SolutionVersionSummary
- Tag
- ThemeGenerationConfig
- TrainingDataConfig
- TunedHPOParams

Tipe data berikut didukung oleh Amazon Personalize Events:

- Action
- ActionInteraction
- Event
- Item
- MetricAttribution
- User

Tipe data berikut ini didukung oleh Amazon Personalize Runtime:

- PredictedAction
- PredictedItem
- Promotion

Amazon Personalize

Tipe data berikut ini didukung oleh Amazon Personalize:

- Algorithm
- AlgorithmImage
- AutoMLConfig
- AutoMLResult
- AutoTrainingConfig
- BatchInferenceJob

- BatchInferenceJobConfig
- BatchInferenceJobInput
- BatchInferenceJobOutput
- BatchInferenceJobSummary
- BatchSegmentJob
- BatchSegmentJobInput
- BatchSegmentJobOutput
- BatchSegmentJobSummary
- Campaign
- CampaignConfig
- CampaignSummary
- CampaignUpdateSummary
- CategoricalHyperParameterRange
- ContinuousHyperParameterRange
- DataDeletionJob
- DataDeletionJobSummary
- Dataset
- DatasetExportJob
- DatasetExportJobOutput
- DatasetExportJobSummary
- DatasetGroup
- DatasetGroupSummary
- DatasetImportJob
- DatasetImportJobSummary
- DatasetSchema
- DatasetSchemaSummary
- DatasetSummary
- DatasetUpdateSummary
- DataSource
- DefaultCategoricalHyperParameterRange

- DefaultContinuousHyperParameterRange
- DefaultHyperParameterRanges
- DefaultIntegerHyperParameterRange
- EventParameters
- EventsConfig
- EventTracker
- EventTrackerSummary
- FeatureTransformation
- FieldsForThemeGeneration
- Filter
- FilterSummary
- HPOConfig
- HPOObjective
- HPOResourceConfig
- HyperParameterRanges
- IntegerHyperParameterRange
- MetricAttribute
- MetricAttribution
- MetricAttributionOutput
- MetricAttributionSummary
- OptimizationObjective
- Recipe
- RecipeSummary
- Recommender
- RecommenderConfig
- RecommenderSummary
- RecommenderUpdateSummary
- S3DataConfig
- Solution
- SolutionConfig

- SolutionSummary
- SolutionUpdateConfig
- SolutionUpdateSummary
- SolutionVersion
- SolutionVersionSummary
- Tag
- ThemeGenerationConfig
- TrainingDataConfig
- TunedHPOParams

Algorithm

Layanan: Amazon Personalize

Menjelaskan algoritma kustom.

Daftar Isi

algorithmArn

Nama Sumber Daya Amazon (ARN) dari algoritma.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

algorithmImage

URI dari wadah Docker untuk gambar algoritma.

Tipe: Objek AlgorithmImage

Wajib: Tidak

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa algoritma dibuat.

Tipe: Timestamp

Wajib: Tidak

defaultHyperParameterRanges

Menentukan hyperparameters default, rentang mereka, dan apakah mereka dapat disetel. Hyperparameter yang dapat disetel dapat menentukan nilainya selama optimasi hyperparameter (HPO).

Tipe: Objek DefaultHyperParameterRanges

Wajib: Tidak

defaultHyperParameters

Menentukan hyperparameters default.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

defaultResourceConfig

Menentukan jumlah maksimum default pekerjaan pelatihan dan pekerjaan pelatihan paralel.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa algoritma terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama algoritma.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

roleArn

Amazon Resource Name (ARN) dari peran.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

trainingInputMode

Mode input pelatihan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

AlgorithmImage

Layanan: Amazon Personalize

Menjelaskan gambar algoritma.

Daftar Isi

dockerURI

URI dari wadah Docker untuk gambar algoritma.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Ya

name

Nama gambar algoritma.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

AutoMLConfig

Layanan: Amazon Personalize

Saat solusi menjalankan AutoML (`performAutoMLtrue` in CreateSolution), Amazon Personalize menentukan resep mana, dari daftar yang ditentukan, mengoptimalkan metrik yang diberikan. Amazon Personalize kemudian menggunakan resep itu untuk solusinya.

Daftar Isi

metricName

Metrik untuk mengoptimalkan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

recipeList

Daftar resep kandidat.

Tipe: Array string

Anggota Array: Jumlah maksimum 100 item.

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

AutoMLResult

Layanan: Amazon Personalize

Saat solusi melakukan AutoML (`performAutoML`true in CreateSolution), tentukan resep yang paling mengoptimalkan metrik yang ditentukan.

Daftar Isi

bestRecipeArn

Nama Sumber Daya Amazon (ARN) dari resep terbaik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

AutoTrainingConfig

Layanan: Amazon Personalize

Konfigurasi pelatihan otomatis untuk digunakan kapan `performAutoTraining` benar.

Daftar Isi

schedulingExpression

Menentukan seberapa sering untuk secara otomatis melatih versi solusi baru. Tentukan ekspresi laju dalam format rate (value unit). Untuk nilai, tentukan angka antara 1 dan 30. Untuk unit, tentukan `day` atau`days`. Misalnya, untuk secara otomatis membuat versi solusi baru setiap 5 hari, tentukan`rate(5 days)`. Defaultnya adalah setiap 7 hari.

Untuk informasi selengkapnya tentang auto training, lihat Membuat dan mengonfigurasi solusi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 16.

Pola: `rate\(\d+ days?\)`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchInferenceJob

Layanan: Amazon Personalize

Berisi informasi tentang pekerjaan inferensi batch.

Daftar Isi

batchInferenceJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan inferensi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

batchInferenceJobConfig

Peta string ke string dari detail konfigurasi pekerjaan inferensi batch.

Tipe: Objek BatchInferenceJobConfig

Wajib: Tidak

batchInferenceJobMode

Modus pekerjaan.

Tipe: String

Nilai yang Valid: `BATCH_INFERENCE | THEME_GENERATION`

Wajib: Tidak

creationDateTime

Waktu di mana pekerjaan inferensi batch dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika pekerjaan inferensi batch gagal, alasan kegagalan.

Tipe: String

Wajib: Tidak

filterArn

ARN dari filter yang digunakan pada pekerjaan inferensi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

jobInput

Jalur Amazon S3 yang mengarah ke data input yang digunakan untuk menghasilkan pekerjaan inferensi batch.

Tipe: Objek BatchInferenceJobInput

Wajib: Tidak

jobName

Nama pekerjaan inferensi batch.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

jobOutput

Bucket Amazon S3 yang berisi data keluaran yang dihasilkan oleh pekerjaan inferensi batch.

Tipe: Objek BatchInferenceJobOutput

Wajib: Tidak

lastUpdatedDateTime

Waktu di mana pekerjaan inferensi batch terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

numResults

Jumlah rekomendasi yang dihasilkan oleh pekerjaan inferensi batch. Nomor ini mencakup pesan kesalahan yang dihasilkan untuk catatan masukan yang gagal.

Tipe: Integer

Wajib: Tidak

roleArn

ARN dari peran Amazon Identity and Access Management (IAM) and Access Management (IAM) yang meminta pekerjaan inferensi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi tempat pekerjaan inferensi batch dibuat.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status pekerjaan inferensi batch. Status adalah salah satu dari nilai berikut:

- MENUNGGU
- SEDANG BERLANGSUNG
- AKTIF
- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

themeGenerationConfig

Pengaturan pembuatan tema pekerjaan.

Tipe: Objek ThemeGenerationConfig

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchInferenceJobConfig

Layanan: Amazon Personalize

Detail konfigurasi pekerjaan inferensi batch.

Daftar Isi

itemExplorationConfig

String ke peta string yang menentukan hyperparameter konfigurasi eksplorasi, termasuk `explorationWeight` dan`explorationItemAgeCutOff`, yang ingin Anda gunakan untuk mengonfigurasi jumlah eksplorasi item yang digunakan Amazon Personalize saat merekomendasikan item. Lihat Personalisasi Pengguna.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchInferenceJobInput

Layanan: Amazon Personalize

Konfigurasi input dari pekerjaan inferensi batch.

Daftar Isi

s3DataSource

URI lokasi Amazon S3 yang berisi data input Anda. Bucket Amazon S3 harus berada di wilayah yang sama dengan titik akhir API yang Anda panggil.

Tipe: Objek S3DataConfig

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchInferenceJobOutput

Layanan: Amazon Personalize

Parameter konfigurasi output dari pekerjaan inferensi batch.

Daftar Isi

s3DataDestination

Informasi tentang bucket Amazon S3 tempat output pekerjaan inferensi batch disimpan.

Tipe: Objek S3DataConfig

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchInferenceJobSummary

Layanan: Amazon Personalize

Versi terpotong dari. BatchInferenceJob ListBatchInferenceJobsOperasi mengembalikan daftar ringkasan pekerjaan inferensi batch.

Daftar Isi

batchInferenceJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan inferensi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

batchInferenceJobMode

Mode pekerjaan.

Tipe: String

Nilai yang Valid: `BATCH_INFERENCE | THEME_GENERATION`

Wajib: Tidak

creationDateTime

Waktu di mana pekerjaan inferensi batch dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika pekerjaan inferensi batch gagal, alasan kegagalan.

Tipe: String

Wajib: Tidak

jobName

Nama pekerjaan inferensi batch.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

lastUpdatedDateTime

Waktu di mana pekerjaan inferensi batch terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

solutionVersionArn

ARN dari versi solusi yang digunakan oleh pekerjaan inferensi batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status pekerjaan inferensi batch. Status adalah salah satu dari nilai berikut:

- MENUNGGU
- SEDANG BERLANGSUNG
- AKTIF
- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchSegmentJob

Layanan: Amazon Personalize

Berisi informasi tentang pekerjaan segmen batch.

Daftar Isi

batchSegmentJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

creationDateTime

Waktu di mana pekerjaan segmen batch dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika pekerjaan segmen batch gagal, alasan kegagalan.

Tipe: String

Wajib: Tidak

filterArn

ARN dari filter yang digunakan pada pekerjaan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

jobInput

Jalur Amazon S3 yang mengarah ke data input yang digunakan untuk menghasilkan pekerjaan segmen batch.

Tipe: Objek BatchSegmentJobInput

Wajib: Tidak

jobName

Nama pekerjaan segmen batch.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

jobOutput

Bucket Amazon S3 yang berisi data keluaran yang dihasilkan oleh pekerjaan segmen batch.

Tipe: Objek BatchSegmentJobOutput

Wajib: Tidak

lastUpdatedDateTime

Waktu di mana pekerjaan segmen batch terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

numResults

Jumlah pengguna yang diprediksi dihasilkan oleh pekerjaan segmen batch untuk setiap baris data input. Jumlah maksimum pengguna per segmen adalah 5 juta.

Tipe: Integer

Wajib: Tidak

roleArn

ARN dari peran Amazon Identity and Access Management (IAM) and Access Management (IAM) yang meminta pekerjaan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi yang digunakan oleh tugas segmen batch untuk menghasilkan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status pekerjaan segmen batch. Status adalah salah satu dari nilai berikut:

- MENUNGGU
- SEDANG BERLANGSUNG
- AKTIF
- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchSegmentJobInput

Layanan: Amazon Personalize

Konfigurasi input dari pekerjaan segmen batch.

Daftar Isi

s3DataSource

Detail konfigurasi bucket input atau output Amazon S3.

Tipe: Objek S3DataConfig

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchSegmentJobOutput

Layanan: Amazon Personalize

Parameter konfigurasi output dari pekerjaan segmen batch.

Daftar Isi

s3DataDestination

Detail konfigurasi bucket input atau output Amazon S3.

Tipe: Objek S3DataConfig

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

BatchSegmentJobSummary

Layanan: Amazon Personalize

Versi tipe data yang terpotong. BatchSegmentJob ListBatchSegmentJobsoperasi mengembalikan daftar ringkasan pekerjaan segmen batch.

Daftar Isi

batchSegmentJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

creationDateTime

Waktu di mana pekerjaan segmen batch dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika pekerjaan segmen batch gagal, alasan kegagalan.

Tipe: String

Wajib: Tidak

jobName

Nama pekerjaan segmen batch.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

lastUpdatedDateTime

Waktu di mana pekerjaan segmen batch terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi yang digunakan oleh tugas segmen batch untuk menghasilkan segmen batch.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status pekerjaan segmen batch. Status adalah salah satu dari nilai berikut:

- MENUNGGU
- SEDANG BERLANGSUNG
- AKTIF
- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

Campaign

Layanan: Amazon Personalize

Objek yang menjelaskan penerapan versi solusi. Untuk informasi selengkapnya tentang kampanye, lihat CreateCampaign.

Daftar Isi

campaignArn

Amazon Resource Name (ARN) dari kampanye.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

campaignConfig

Detail konfigurasi kampanye.

Tipe: Objek CampaignConfig

Wajib: Tidak

creationDateTime

Tanggal dan waktu (dalam format Unix) kampanye dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika kampanye gagal, alasan di balik kegagalan.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam format Unix) kampanye terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

latestCampaignUpdate

Menyediakan ringkasan properti pembaruan kampanye. Untuk daftar lengkap, hubungi DescribeCampaignAPI.

Tipe: Objek CampaignUpdateSummary

Wajib: Tidak

minProvisionedTPS

Menentukan transaksi minimum yang disediakan (rekomendasi) per detik yang diminta. Tinggi `minProvisionedTPS` akan meningkatkan tagihan Anda. Sebaiknya mulai dengan 1 untuk `minProvisionedTPS` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minProvisionedTPS` seperlunya.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1.

Wajib: Tidak

name

Nama kampanye.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi yang digunakan kampanye.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status kampanye.

Kampanye dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CampaignConfig

Layanan: Amazon Personalize

Detail konfigurasi kampanye.

Daftar Isi

enableMetadataWithRecommendations

Apakah metadata dengan rekomendasi diaktifkan untuk kampanye. Jika diaktifkan, Anda dapat menentukan kolom dari kumpulan data Item dalam permintaan rekomendasi. Amazon Personalize mengembalikan data ini untuk setiap item dalam respons rekomendasi. Untuk informasi tentang mengaktifkan metadata untuk kampanye, lihat Mengaktifkan metadata dalam rekomendasi untuk kampanye.

Jika Anda mengaktifkan metadata dalam rekomendasi, Anda akan dikenakan biaya tambahan. Untuk informasi selengkapnya, lihat Harga Amazon Personalisasi.

Tipe: Boolean

Wajib: Tidak

itemExplorationConfig

Menentukan hiperparameter konfigurasi eksplorasi, termasuk `explorationWeight` dan `explorationItemAgeCutOff` yang ingin Anda gunakan untuk mengonfigurasi jumlah eksplorasi item yang digunakan Amazon Personalize saat merekomendasikan item. Berikan `itemExplorationConfig` data hanya jika solusi Anda menggunakan resep Personalisasi Pengguna.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

syncWithLatestSolutionVersion

Apakah kampanye diperbarui secara otomatis untuk menggunakan versi solusi terbaru (model terlatih) dari solusi. Jika Anda menentukan`True`, Anda harus menentukan ARN solusi Anda untuk

parameter. `SolutionVersionArn` Itu harus dalam `SolutionArn/$LATEST` format. Defaultnya adalah `False` dan Anda harus memperbarui kampanye secara manual untuk menerapkan versi solusi terbaru.

Untuk informasi selengkapnya tentang pembaruan kampanye otomatis, lihat Mengaktifkan pembaruan kampanye otomatis.

Tipe: Boolean

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CampaignSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti kampanye. Untuk daftar lengkap, hubungi DescribeCampaignAPI.

Daftar Isi

campaignArn

Amazon Resource Name (ARN) dari kampanye.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

creationDateTime

Tanggal dan waktu (dalam waktu Unix) kampanye dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika kampanye gagal, alasan di balik kegagalan.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) kampanye terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama kampanye.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status kampanye.

Kampanye dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CampaignUpdateSummary

Layanan: Amazon Personalize

Menyediakan ringkasan properti pembaruan kampanye. Untuk daftar lengkap, hubungi DescribeCampaignAPI.

Daftar Isi

campaignConfig

Detail konfigurasi kampanye.

Tipe: Objek CampaignConfig

Wajib: Tidak

creationDateTime

Tanggal dan waktu (dalam waktu Unix) pembaruan kampanye dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika update kampanye gagal, alasan di balik kegagalan.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) pembaruan kampanye terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

minProvisionedTPS

Menentukan transaksi minimum yang disediakan (rekomendasi) per detik yang akan didukung Amazon Personalize.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1.

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi yang diterapkan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status pembaruan kampanye.

Pembaruan kampanye dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CategoricalHyperParameterRange

Layanan: Amazon Personalize

Memberikan nama dan rentang hyperparameter kategoris.

Daftar Isi

name

Nama hyperparameter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

values

Daftar kategori untuk hyperparameter.

Tipe: Array string

Anggota Array: Jumlah maksimum 100 item.

Panjang Batasan: Panjang maksimum 1000.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

ContinuousHyperParameterRange

Layanan: Amazon Personalize

Memberikan nama dan rentang hyperparameter kontinu.

Daftar Isi

maxValue

Nilai maksimum yang diijinkan untuk hyperparameter.

Tipe: Ganda

Rentang Valid: Nilai minimum -1000000.

Wajib: Tidak

minValue

Nilai minimum yang diijinkan untuk hyperparameter.

Tipe: Ganda

Rentang Valid: Nilai minimum -1000000.

Wajib: Tidak

name

Nama hyperparameter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

DataDeletionJob

Layanan: Amazon Personalize

Menjelaskan pekerjaan yang menghapus semua referensi ke pengguna tertentu dari grup kumpulan data Amazon Personalize dalam batch. Untuk informasi tentang membuat pekerjaan penghapusan data, lihat Menghapus pengguna.

Daftar Isi

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari pekerjaan penghapusan data.

Tipe: Timestamp

Wajib: Tidak

dataDeletionJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan penghapusan data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data tempat tugas menghapus catatan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

dataSource

Menjelaskan sumber data yang berisi data yang akan diunggah ke kumpulan data, atau daftar catatan yang akan dihapus dari Amazon Personalisasi.

Tipe: Objek DataSource

Wajib: Tidak

failureReason

Jika pekerjaan penghapusan data gagal, berikan alasannya.

Tipe: String

Wajib: Tidak

jobName

Nama pekerjaan penghapusan data.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) pekerjaan penghapusan data terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

numDeleted

Jumlah catatan yang dihapus oleh pekerjaan SELESAI.

Tipe: Integer

Wajib: Tidak

roleArn

Nama Sumber Daya Amazon (ARN) dari peran IAM yang memiliki izin untuk membaca dari sumber data Amazon S3.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Tidak

status

Status pekerjaan penghapusan data.

Pekerjaan penghapusan data dapat memiliki salah satu status berikut:

- TERTUNDA > IN_PROGRESS> SELESAI -atau- GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DataDeletionJobSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti pekerjaan penghapusan data. Untuk daftar lengkap, hubungi operasi DescribeDataDeletionJobAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari pekerjaan penghapusan data.

Tipe: Timestamp

Wajib: Tidak

dataDeletionJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan penghapusan data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data mengelompokkan catatan pekerjaan yang dihapus.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

failureReason

Jika pekerjaan penghapusan data gagal, berikan alasannya.

Tipe: String

Wajib: Tidak

jobName

Nama pekerjaan penghapusan data.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) pekerjaan penghapusan data terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

status

Status pekerjaan penghapusan data.

Pekerjaan penghapusan data dapat memiliki salah satu status berikut:

- TERTUNDA > IN_PROGRESS> SELESAI -atau- GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

Dataset

Layanan: Amazon Personalize

Menyediakan metadata untuk kumpulan data.

Daftar Isi

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari dataset.

Tipe: Timestamp

Wajib: Tidak

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data yang Anda inginkan untuk metadata.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

datasetType

Salah satu nilai berikut:

- Interaksi
- Item

- Pengguna
- Tindakan
- Action_Interactions

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

lastUpdatedDateTime

Cap waktu yang menunjukkan kapan kumpulan data diperbarui.

Tipe: Timestamp

Wajib: Tidak

latestDatasetUpdate

Menjelaskan pembaruan terbaru pada kumpulan data.

Tipe: Objek DatasetUpdateSummary

Wajib: Tidak

name

Nama set data.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

schemaArn

ARN dari skema terkait.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status dataset.

Dataset dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

trackingId

ID pelacak peristiwa untuk kumpulan data interaksi Tindakan. Anda menentukan ID pelacak dalam operasi `PutActionInteractions` API. Amazon Personalize menggunakannya untuk mengarahkan data baru ke kumpulan data interaksi Tindakan di grup kumpulan data Anda.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetExportJob

Layanan: Amazon Personalize

Menjelaskan pekerjaan yang mengekspor kumpulan data ke bucket Amazon S3. Untuk informasi selengkapnya, lihat CreateDatasetExportJob.

Pekerjaan ekspor dataset dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Daftar Isi

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari pekerjaan ekspor dataset.

Tipe: Timestamp

Wajib: Tidak

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data yang akan diekspor.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

datasetExportJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan ekspor dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

failureReason

Jika pekerjaan ekspor kumpulan data gagal, berikan alasannya.

Tipe: String

Wajib: Tidak

ingestionMode

Data yang akan diekspor, berdasarkan cara Anda mengimpor data. Anda dapat memilih untuk mengekspor `BULK` data yang diimpor menggunakan pekerjaan impor kumpulan data, `PUT` data yang diimpor secara bertahap (menggunakan konsol, PutUsers dan PutItems operasi) PutEvents, atau `ALL` untuk kedua jenis tersebut. Nilai default-nya adalah `PUT`.

Tipe: String

Nilai yang Valid: `BULK | PUT | ALL`

Wajib: Tidak

jobName

Nama pekerjaan ekspor.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

jobOutput

Jalur ke bucket Amazon S3 tempat output pekerjaan disimpan. Misalnya:

`s3://bucket-name/folder-name/`

Tipe: Objek DatasetExportJobOutput

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) status pekerjaan ekspor dataset terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

roleArn

Nama Sumber Daya Amazon (ARN) dari peran layanan IAM yang memiliki izin untuk menambahkan data ke bucket Amazon S3 keluaran Anda.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status pekerjaan ekspor dataset.

Pekerjaan ekspor dataset dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetExportJobOutput

Layanan: Amazon Personalize

Parameter konfigurasi output dari pekerjaan ekspor dataset.

Daftar Isi

s3DataDestination

Detail konfigurasi bucket input atau output Amazon S3.

Tipe: Objek S3DataConfig

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetExportJobSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti pekerjaan ekspor dataset. Untuk daftar lengkap, hubungi DescribeDatasetExportJobAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa pekerjaan ekspor dataset dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetExportJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan ekspor dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

failureReason

Jika pekerjaan ekspor dataset gagal, alasan di balik kegagalan.

Tipe: String

Wajib: Tidak

jobName

Nama pekerjaan ekspor dataset.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) status pekerjaan ekspor dataset terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

status

Status pekerjaan ekspor dataset.

Pekerjaan ekspor dataset dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetGroup

Layanan: Amazon Personalize

Grup dataset adalah kumpulan kumpulan data terkait (Interaksi item, Pengguna, Item, Tindakan, Interaksi aksi). Anda membuat grup kumpulan data dengan menelepon CreateDatasetGroup. Anda kemudian membuat kumpulan data dan menambahkannya ke grup kumpulan data dengan menelepon. CreateDataset Grup dataset digunakan untuk membuat dan melatih solusi dengan menelepon CreateSolution. Grup dataset hanya dapat berisi satu dari setiap jenis dataset.

Anda dapat menentukan kunci AWS Key Management Service (KMS) untuk mengenkripsi kumpulan data dalam grup.

Daftar Isi

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari grup dataset.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

domain

Domain dari grup dataset Domain.

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Wajib: Tidak

failureReason

Jika membuat grup kumpulan data gagal, berikan alasannya.

Tipe: String

Wajib: Tidak

kmsKeyArn

Nama Sumber Daya Amazon (ARN) dari kunci AWS Key Management Service (KMS) yang digunakan untuk mengenkripsi kumpulan data.

Tipe: String

Batasan Panjang: Panjang maksimum 2048.

Pola: `arn:aws.*:kms:.*:[0-9]{12}:key/.*`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu pembaruan terakhir (dalam waktu Unix) dari grup kumpulan data.

Tipe: Timestamp

Wajib: Tidak

name

Nama grup dataset.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

roleArn

ARN dari peran AWS Identity and Access Management (IAM) yang memiliki izin untuk mengakses kunci AWS Key Management Service (KMS). Menyediakan peran IAM hanya valid ketika juga menentukan kunci KMS.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Tidak

status

Status grup dataset saat ini.

Grup kumpulan data dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetGroupSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti grup dataset. Untuk daftar lengkap, hubungi DescribeDatasetGroupAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa grup dataset dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

domain

Domain dari grup dataset Domain.

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Wajib: Tidak

failureReason

Jika membuat grup dataset gagal, alasan di balik kegagalan.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa grup dataset terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama grup dataset.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status grup dataset.

Grup kumpulan data dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetImportJob

Layanan: Amazon Personalize

Menjelaskan pekerjaan yang mengimpor data pelatihan dari sumber data (bucket Amazon S3) ke kumpulan data Amazon Personalize. Untuk informasi selengkapnya, lihat CreateDatasetImportJob.

Pekerjaan impor dataset dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Daftar Isi

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari pekerjaan impor dataset.

Tipe: Timestamp

Wajib: Tidak

datasetArn

Nama Sumber Daya Amazon (ARN) dari kumpulan data yang menerima data yang diimpor.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

datasetImportJobArn

ARN dari pekerjaan impor dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

dataSource

Bucket Amazon S3 yang berisi data pelatihan untuk diimpor.

Tipe: Objek DataSource

Wajib: Tidak

failureReason

Jika pekerjaan impor kumpulan data gagal, berikan alasannya.

Tipe: String

Wajib: Tidak

importMode

Mode impor yang digunakan oleh pekerjaan impor dataset untuk mengimpor catatan baru.

Tipe: String

Nilai yang Valid: `FULL | INCREMENTAL`

Wajib: Tidak

jobName

Nama pekerjaan impor.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) dataset terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

publishAttributionMetricsToS3

Apakah lowongan memublikasikan metrik ke Amazon S3 untuk atribusi metrik.

Tipe: Boolean

Wajib: Tidak

roleArn

ARN dari peran IAM yang memiliki izin untuk membaca dari sumber data Amazon S3.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status pekerjaan impor dataset.

Pekerjaan impor dataset dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetImportJobSummary

Layanan: Amazon Personalize

Menyediakan ringkasan properti pekerjaan impor dataset. Untuk daftar lengkap, hubungi DescribeDatasetImportJobAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa pekerjaan impor dataset dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetImportJobArn

Nama Sumber Daya Amazon (ARN) dari pekerjaan impor dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

failureReason

Jika pekerjaan impor dataset gagal, alasan di balik kegagalan.

Tipe: String

Wajib: Tidak

importMode

Mode impor pekerjaan impor dataset yang digunakan untuk memperbarui data dalam dataset. Untuk informasi selengkapnya lihat Memperbarui data massal yang ada.

Tipe: String

Nilai yang Valid: `FULL | INCREMENTAL`

Wajib: Tidak

jobName

Nama pekerjaan impor dataset.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) status pekerjaan impor dataset terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

status

Status pekerjaan impor dataset.

Pekerjaan impor dataset dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetSchema

Layanan: Amazon Personalize

Menjelaskan skema untuk kumpulan data. Untuk informasi selengkapnya tentang skema, lihat CreateSchema.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) skema dibuat.

Tipe: Timestamp

Wajib: Tidak

domain

Domain skema yang Anda buat untuk kumpulan data dalam grup dataset Domain.

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa skema terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama skema.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

schema

Skema.

Tipe: String

Kendala Panjang: Panjang maksimum 20000.

Wajib: Tidak

schemaArn

Amazon Resource Name (ARN) dari skema.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetSchemaSummary

Layanan: Amazon Personalize

Menyediakan ringkasan properti skema dataset. Untuk daftar lengkap, hubungi DescribeSchemaAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) skema dibuat.

Tipe: Timestamp

Wajib: Tidak

domain

Domain skema yang Anda buat untuk kumpulan data dalam grup dataset Domain.

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa skema terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama skema.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

schemaArn

Amazon Resource Name (ARN) dari skema.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetSummary

Layanan: Amazon Personalize

Menyediakan ringkasan properti dari dataset. Untuk daftar lengkap, hubungi DescribeDatasetAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa dataset dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetArn

Amazon Resource Name (ARN) dari set data.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

datasetType

Jenis dataset. Salah satu nilai berikut:

- Interaksi
- Item
- Pengguna
- Event-Interaksi

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa dataset terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama set data.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status dataset.

Dataset dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DatasetUpdateSummary

Layanan: Amazon Personalize

Menjelaskan pembaruan pada kumpulan data.

Daftar Isi

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari pembaruan dataset.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika memperbarui kumpulan data gagal, berikan alasannya.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu pembaruan terakhir (dalam waktu Unix) dari kumpulan data.

Tipe: Timestamp

Wajib: Tidak

schemaArn

Nama Sumber Daya Amazon (ARN) dari skema yang menggantikan skema kumpulan data sebelumnya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status pembaruan dataset.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DataSource

Layanan: Amazon Personalize

Menjelaskan sumber data yang berisi data yang akan diunggah ke kumpulan data, atau daftar catatan yang akan dihapus dari Amazon Personalisasi.

Daftar Isi

dataLocation

Untuk pekerjaan impor kumpulan data, jalur ke bucket Amazon S3 tempat data yang ingin Anda unggah ke kumpulan data disimpan. Untuk pekerjaan penghapusan data, jalur ke bucket Amazon S3 yang menyimpan daftar catatan yang akan dihapus.

Misalnya:

`s3://bucket-name/folder-name/fileName.csv`

Jika file CSV Anda berada di folder di bucket Amazon S3 dan Anda ingin pekerjaan impor atau pekerjaan penghapusan data mempertimbangkan beberapa file, Anda dapat menentukan jalur ke folder tersebut. Dengan pekerjaan penghapusan data, Amazon Personalize menggunakan semua file di folder dan sub folder apa pun. Gunakan sintaks berikut dengan nama folder `/` setelah:

`s3://bucket-name/folder-name/`

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `(s3|http|https)://.+`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DefaultCategoricalHyperParameterRange

Layanan: Amazon Personalize

Menyediakan nama dan rentang default dari hyperparameter kategoris dan apakah hyperparameter dapat disetel. Hyperparameter yang dapat disetel dapat menentukan nilainya selama optimasi hyperparameter (HPO).

Daftar Isi

isTunable

Apakah hyperparameter dapat disetel.

Tipe: Boolean

Wajib: Tidak

name

Nama hyperparameter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

values

Daftar kategori untuk hyperparameter.

Tipe: Array string

Anggota Array: Jumlah maksimum 100 item.

Panjang Batasan: Panjang maksimum 1000.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DefaultContinuousHyperParameterRange

Layanan: Amazon Personalize

Menyediakan nama dan rentang default dari hyperparameter kontinu dan apakah hyperparameter dapat disetel. Hyperparameter yang dapat disetel dapat menentukan nilainya selama optimasi hyperparameter (HPO).

Daftar Isi

isTunable

Apakah hyperparameter dapat disetel.

Tipe: Boolean

Wajib: Tidak

maxValue

Nilai maksimum yang diijinkan untuk hyperparameter.

Tipe: Ganda

Rentang Valid: Nilai minimum -1000000.

Wajib: Tidak

minValue

Nilai minimum yang diijinkan untuk hyperparameter.

Tipe: Ganda

Rentang Valid: Nilai minimum -1000000.

Wajib: Tidak

name

Nama hyperparameter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DefaultHyperParameterRanges

Layanan: Amazon Personalize

Menentukan hyperparameters dan rentang default mereka. Hyperparameter dapat berupa kategoris, kontinu, atau bernilai bilangan bulat.

Daftar Isi

categoricalHyperParameterRanges

Hiperparameter kategoris dan rentang defaultnya.

Tipe: Array objek DefaultCategoricalHyperParameterRange

Anggota Array: Jumlah maksimum 100 item.

Wajib: Tidak

continuousHyperParameterRanges

Hyperparameter kontinu dan rentang defaultnya.

Tipe: Array objek DefaultContinuousHyperParameterRange

Anggota Array: Jumlah maksimum 100 item.

Wajib: Tidak

integerHyperParameterRanges

Hyperparameter bernilai bilangan bulat dan rentang defaultnya.

Tipe: Array objek DefaultIntegerHyperParameterRange

Anggota Array: Jumlah maksimum 100 item.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

DefaultIntegerHyperParameterRange

Layanan: Amazon Personalize

Menyediakan nama dan rentang default dari hyperparameter bernilai integer dan apakah hyperparameter dapat disetel. Hyperparameter yang dapat disetel dapat menentukan nilainya selama optimasi hyperparameter (HPO).

Daftar Isi

isTunable

Menunjukkan apakah hyperparameter dapat disetel.

Tipe: Boolean

Wajib: Tidak

maxValue

Nilai maksimum yang diijinkan untuk hyperparameter.

Jenis: Integer

Rentang Valid: Nilai maksimum 1000000.

Wajib: Tidak

minValue

Nilai minimum yang diijinkan untuk hyperparameter.

Jenis: Integer

Rentang Valid: Nilai minimum -1000000.

Wajib: Tidak

name

Nama hyperparameter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

EventParameters

Layanan: Amazon Personalize

Menjelaskan parameter peristiwa, yang digunakan dalam pembuatan solusi.

Daftar Isi

eventType

Nama jenis acara yang akan dipertimbangkan untuk pembuatan solusi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

eventValueThreshold

Ambang batas jenis acara. Hanya peristiwa dengan nilai lebih besar atau sama dengan ambang batas ini yang akan dipertimbangkan untuk pembuatan solusi.

Tipe: Ganda

Wajib: Tidak

weight

Berat jenis acara. Bobot yang lebih tinggi berarti lebih penting dari jenis acara untuk solusi yang dibuat.

Tipe: Ganda

Rentang yang Valid: Nilai minimum 0. Nilai maksimum 1.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

EventsConfig

Layanan: Amazon Personalize

Menjelaskan konfigurasi peristiwa, yang digunakan dalam pembuatan solusi.

Daftar Isi

eventParametersList

Daftar parameter acara, yang mencakup jenis acara dan ambang batas nilai acara dan bobotnya.

Tipe: Array objek EventParameters

Anggota Array: Jumlah maksimum 10 item.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

EventTracker

Layanan: Amazon Personalize

Memberikan informasi tentang pelacak acara.

Daftar Isi

accountId

AWS Akun yang memiliki pelacak acara.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

creationDateTime

Tanggal dan waktu (dalam format Unix) bahwa pelacak acara dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data yang menerima data peristiwa.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

eventTrackerArn

ARN dari pelacak acara.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa pelacak acara terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama pelacak acara.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status pelacak acara.

Pelacak peristiwa dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

trackingId

ID pelacak acara. Sertakan ID ini dalam permintaan ke PutEventsAPI.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

EventTrackerSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti pelacak peristiwa. Untuk daftar lengkap, hubungi DescribeEventTrackerAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa pelacak acara dibuat.

Tipe: Timestamp

Wajib: Tidak

eventTrackerArn

Nama Sumber Daya Amazon (ARN) dari pelacak acara.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa pelacak acara terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama pelacak acara.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status pelacak acara.

Pelacak peristiwa dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FeatureTransformation

Layanan: Amazon Personalize

Menyediakan informasi transformasi fitur. Transformasi fitur adalah proses memodifikasi data input mentah menjadi bentuk yang lebih cocok untuk pelatihan model.

Daftar Isi

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari transformasi fitur.

Tipe: Timestamp

Wajib: Tidak

defaultParameters

Menyediakan parameter default untuk transformasi fitur.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

featureTransformationArn

Nama Sumber Daya Amazon (ARN) dari objek. FeatureTransformation

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu pembaruan terakhir (dalam waktu Unix) dari transformasi fitur.

Tipe: Timestamp

Wajib: Tidak

name

Nama transformasi fitur.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status transformasi fitur.

Transformasi fitur dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FieldsForThemeGeneration

Layanan: Amazon Personalize

Peta string ke string dari detail konfigurasi untuk pembuatan tema.

Daftar Isi

itemName

Nama kolom kumpulan data Item yang menyimpan nama setiap item dalam kumpulan data.

Tipe: String

Kendala Panjang: Panjang maksimum 150.

Pola: `[A-Za-z_][A-Za-z\d_]*`

Diperlukan: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Filter

Layanan: Amazon Personalize

Berisi informasi tentang filter rekomendasi, termasuk ARN, status, dan ekspresi filternya.

Daftar Isi

creationDateTime

Waktu di mana filter dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

ARN dari grup dataset tempat filter berada.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

failureReason

Jika filter gagal, alasan kegagalannya.

Tipe: String

Wajib: Tidak

filterArn

ARN dari filter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

filterExpression

Menentukan jenis interaksi item untuk menyaring hasil rekomendasi. Ekspresi filter harus mengikuti aturan format tertentu. Untuk informasi tentang struktur ekspresi filter dan sintaks, lihat Menyaring ekspresi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 2500.

Wajib: Tidak

lastUpdatedDateTime

Waktu di mana filter terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama filter.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status filter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

FilterSummary

Layanan: Amazon Personalize

Ringkasan singkat dari atribut filter.

Daftar Isi

creationDateTime

Waktu di mana filter dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

ARN dari grup dataset tempat filter berada.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

failureReason

Jika filter gagal, alasan kegagalan.

Tipe: String

Wajib: Tidak

filterArn

ARN dari filter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

lastUpdatedDateTime

Waktu di mana filter terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama filter.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status filter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

HPOConfig

Layanan: Amazon Personalize

Menjelaskan properti untuk optimasi hyperparameter (HPO).

Daftar Isi

algorithmHyperParameterRanges

Hiperparameter dan rentang yang diijinkan.

Tipe: Objek HyperParameterRanges

Wajib: Tidak

hpoObjective

Metrik untuk dioptimalkan selama HPO.

> **Note**
>
> Amazon Personalize tidak mendukung konfigurasi `hpoObjective` saat ini.

Tipe: Objek HPOObjective

Wajib: Tidak

hpoResourceConfig

Menjelaskan konfigurasi sumber daya untuk HPO.

Tipe: Objek HPOResourceConfig

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

HPOObjective

Layanan: Amazon Personalize

Metrik untuk dioptimalkan selama optimasi hyperparameter (HPO).

> **Note**
>
> Amazon Personalize tidak mendukung konfigurasi `hpoObjective` saat ini.

Daftar Isi

metricName

Nama metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

metricRegex

Ekspresi reguler untuk menemukan metrik dalam log pekerjaan pelatihan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

type

Jenis metrik. Nilai yang valid adalah `Maximize` dan `Minimize`.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

HPOResourceConfig

Layanan: Amazon Personalize

Menjelaskan konfigurasi sumber daya untuk optimasi hyperparameter (HPO).

Daftar Isi

maxNumberOfTrainingJobs

Jumlah maksimum pekerjaan pelatihan saat Anda membuat versi solusi. Nilai maksimum untuk `maxNumberOfTrainingJobs` adalah`40`.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

maxParallelTrainingJobs

Jumlah maksimum pekerjaan pelatihan paralel saat Anda membuat versi solusi. Nilai maksimum untuk `maxParallelTrainingJobs` adalah`10`.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

HyperParameterRanges

Layanan: Amazon Personalize

Menentukan hyperparameters dan rentang mereka. Hyperparameter dapat berupa kategoris, kontinu, atau bernilai bilangan bulat.

Daftar Isi

categoricalHyperParameterRanges

Hiperparameter kategoris dan rentangnya.

Tipe: Array objek CategoricalHyperParameterRange

Anggota Array: Jumlah maksimum 100 item.

Wajib: Tidak

continuousHyperParameterRanges

Hiperparameter kontinu dan rentangnya.

Tipe: Array objek ContinuousHyperParameterRange

Anggota Array: Jumlah maksimum 100 item.

Wajib: Tidak

integerHyperParameterRanges

Hiperparameter bernilai bilangan bulat dan rentangnya.

Tipe: Array objek IntegerHyperParameterRange

Anggota Array: Jumlah maksimum 100 item.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

IntegerHyperParameterRange

Layanan: Amazon Personalize

Memberikan nama dan rentang hyperparameter bernilai bilangan bulat.

Daftar Isi

maxValue

Nilai maksimum yang diijinkan untuk hyperparameter.

Jenis: Integer

Rentang Valid: Nilai maksimum 1000000.

Wajib: Tidak

minValue

Nilai minimum yang diijinkan untuk hyperparameter.

Jenis: Integer

Rentang Valid: Nilai minimum -1000000.

Wajib: Tidak

name

Nama hyperparameter.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

MetricAttribute

Layanan: Amazon Personalize

Berisi informasi tentang metrik yang dilaporkan oleh atribusi metrik. Untuk informasi selengkapnya, lihat Mengukur dampak rekomendasi.

Daftar Isi

eventType

Jenis peristiwa metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Ya

expression

Ekspresi atribut. Fungsi yang tersedia adalah `SUM()` atau`SAMPLECOUNT()`. Untuk fungsi SUM (), berikan jenis kumpulan data (baik Interaksi atau Item) dan kolom untuk dijumlahkan sebagai parameter. Misalnya SUM (Items.price).

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Ya

metricName

Nama metrik. Nama ini membantu Anda mengidentifikasi metrik di Amazon CloudWatch atau Amazon S3.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MetricAttribution

Layanan: Amazon Personalize

Berisi informasi tentang atribusi metrik. Atribusi metrik membuat laporan tentang data yang Anda impor ke Amazon Personalize. Bergantung pada cara Anda mengimpor data, Anda dapat melihat laporan di Amazon CloudWatch atau Amazon S3. Untuk informasi selengkapnya, lihat Mengukur dampak rekomendasi.

Daftar Isi

creationDateTime

Waktu pembuatan atribusi metrik.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

Kumpulan data atribusi metrik mengelompokkan Amazon Resource Name (ARN).

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

failureReason

Alasan kegagalan atribusi metrik.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Waktu tanggal terakhir yang diperbarui atribusi metrik.

Tipe: Timestamp

Wajib: Tidak

metricAttributionArn

Nama Sumber Daya Amazon (ARN) atribusi metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

metricsOutputConfig

Konfigurasi keluaran atribusi metrik.

Tipe: Objek MetricAttributionOutput

Wajib: Tidak

name

Nama atribusi metrik.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status atribusi metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MetricAttributionOutput

Layanan: Amazon Personalize

Rincian konfigurasi keluaran untuk atribusi metrik.

Daftar Isi

roleArn

Nama Sumber Daya Amazon (ARN) dari peran layanan IAM yang memiliki izin untuk menambahkan data ke bucket Amazon S3 keluaran Anda dan menambahkan metrik ke Amazon. CloudWatch Untuk informasi selengkapnya, lihat Mengukur dampak rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):iam::\d{12}:role/?[a-zA-Z_0-9+=,.@\-_/]+`

Wajib: Ya

s3DataDestination

Detail konfigurasi bucket input atau output Amazon S3.

Tipe: Objek S3DataConfig

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

MetricAttributionSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti atribusi metrik. Untuk daftar lengkap, hubungi DescribeMetricAttribution.

Daftar Isi

creationDateTime

Waktu pembuatan atribusi metrik.

Tipe: Timestamp

Wajib: Tidak

failureReason

Alasan kegagalan atribusi metrik.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Waktu tanggal terakhir yang diperbarui atribusi metrik.

Tipe: Timestamp

Wajib: Tidak

metricAttributionArn

Nama Sumber Daya Amazon (ARN) atribusi metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

name

Nama atribusi metrik.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

status

Status atribusi metrik.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

OptimizationObjective

Layanan: Amazon Personalize

Menjelaskan tujuan tambahan untuk solusi, seperti memaksimalkan menit streaming atau meningkatkan pendapatan. Untuk informasi selengkapnya lihat Mengoptimalkan solusi.

Daftar Isi

itemAttribute

Kolom metadata numerik dalam kumpulan data Item yang terkait dengan tujuan pengoptimalan. Misalnya, VIDEO_LENGTH (untuk memaksimalkan menit streaming), atau PRICE (untuk memaksimalkan pendapatan).

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 150.

Wajib: Tidak

objectiveSensitivity

Menentukan bagaimana Amazon Personalize menyeimbangkan pentingnya tujuan pengoptimalan Anda versus relevansi.

Tipe: String

Nilai yang Valid: `LOW | MEDIUM | HIGH | OFF`

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Recipe

Layanan: Amazon Personalize

Memberikan informasi tentang resep. Setiap resep menyediakan algoritme yang digunakan Amazon Personalize dalam pelatihan model saat Anda menggunakan operasi. CreateSolution

Daftar Isi

algorithmArn

Nama Sumber Daya Amazon (ARN) dari algoritme yang digunakan Amazon Personalize untuk melatih model.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

creationDateTime

Tanggal dan waktu (dalam format Unix) resep dibuat.

Tipe: Timestamp

Wajib: Tidak

description

Deskripsi resep.

Tipe: String

Wajib: Tidak

featureTransformationArn

ARN objek. FeatureTransformation

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam format Unix) resep terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama resepnya.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

recipeArn

Nama Sumber Daya Amazon (ARN) dari resepnya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

recipeType

Salah satu nilai berikut:

- PERSONALIZED_RANKING
- RELATED_ITEMS
- USER_PERSONALISASI

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

status

Status resepnya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RecipeSummary

Layanan: Amazon Personalize

Memberikan ringkasan sifat-sifat resep. Untuk daftar lengkap, hubungi DescribeRecipeAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) resep dibuat.

Tipe: Timestamp

Wajib: Tidak

domain

Domain resep (jika resepnya adalah kasus penggunaan grup kumpulan data Domain).

Tipe: String

Nilai yang Valid: `ECOMMERCE | VIDEO_ON_DEMAND`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) resep terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama resepnya.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

recipeArn

Nama Sumber Daya Amazon (ARN) dari resepnya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status resepnya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Recommender

Layanan: Amazon Personalize

Menjelaskan generator rekomendasi untuk grup dataset Domain. Anda membuat pemberi rekomendasi dalam grup kumpulan data Domain untuk kasus penggunaan domain tertentu (resep domain), dan menentukan pemberi rekomendasi dalam permintaan. GetRecommendations

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam format Unix) bahwa pemberi rekomendasi dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset Domain yang berisi pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

failureReason

Jika seorang pemberi rekomendasi gagal, alasan di balik kegagalan tersebut.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam format Unix) bahwa pemberi rekomendasi terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

latestRecommenderUpdate

Memberikan ringkasan pembaruan terbaru kepada pemberi rekomendasi.

Tipe: Objek RecommenderUpdateSummary

Wajib: Tidak

modelMetrics

Menyediakan metrik evaluasi yang membantu Anda menentukan kinerja pemberi rekomendasi. Untuk informasi selengkapnya, lihat Mengevaluasi pemberi rekomendasi.

Tipe: Peta string ke ganda

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Wajib: Tidak

name

Nama pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

recipeArn

Nama Sumber Daya Amazon (ARN) dari resep (Kasus penggunaan grup kumpulan data Domain) tempat pemberi rekomendasi dibuat.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

recommenderConfig

Detail konfigurasi dari pemberi rekomendasi.

Tipe: Objek RecommenderConfig

Wajib: Tidak

status

Status pemberi rekomendasi.

Seorang pemberi rekomendasi dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- STOP PENDING > STOP IN_PROGRESS > TIDAK AKTIF> MULAI TERTUNDA > MULAI IN_PROGRESS > AKTIF
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

RecommenderConfig

Layanan: Amazon Personalize

Detail konfigurasi dari pemberi rekomendasi.

Daftar Isi

enableMetadataWithRecommendations

Apakah metadata dengan rekomendasi diaktifkan untuk pemberi rekomendasi. Jika diaktifkan, Anda dapat menentukan kolom dari kumpulan data Item dalam permintaan rekomendasi. Amazon Personalize mengembalikan data ini untuk setiap item dalam respons rekomendasi. Untuk informasi tentang mengaktifkan metadata untuk pemberi rekomendasi, lihat Mengaktifkan metadata dalam rekomendasi untuk pemberi rekomendasi.

Jika Anda mengaktifkan metadata dalam rekomendasi, Anda akan dikenakan biaya tambahan. Untuk informasi selengkapnya, lihat Harga Amazon Personalisasi.

Tipe: Boolean

Wajib: Tidak

itemExplorationConfig

Menentukan hiperparameter konfigurasi eksplorasi, termasuk `explorationWeight` dan `explorationItemAgeCutOff` yang ingin Anda gunakan untuk mengonfigurasi jumlah eksplorasi item yang digunakan Amazon Personalize saat merekomendasikan item. Berikan `itemExplorationConfig` data hanya jika pemberi rekomendasi Anda menghasilkan rekomendasi yang dipersonalisasi untuk pengguna (bukan item populer atau item serupa).

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

minRecommendationRequestsPerSecond

Menentukan permintaan rekomendasi minimum yang disediakan per detik yang akan didukung Amazon Personalize. Tinggi `minRecommendationRequestsPerSecond` akan meningkatkan

tagihan Anda. Sebaiknya mulai dengan 1 untuk `minRecommendationRequestsPerSecond` (default). Lacak penggunaan Anda menggunakan CloudWatch metrik Amazon, dan tingkatkan `minRecommendationRequestsPerSecond` seperlunya.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1.

Wajib: Tidak

trainingDataConfig

Menentukan konfigurasi data pelatihan yang akan digunakan saat membuat pemberi rekomendasi domain.

Tipe: Objek TrainingDataConfig

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RecommenderSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti pemberi rekomendasi.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam format Unix) bahwa pemberi rekomendasi dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset Domain yang berisi pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam format Unix) bahwa pemberi rekomendasi terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

recipeArn

Nama Sumber Daya Amazon (ARN) dari resep (Kasus penggunaan grup kumpulan data Domain) tempat pemberi rekomendasi dibuat.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

recommenderArn

Nama Sumber Daya Amazon (ARN) dari pemberi rekomendasi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

recommenderConfig

Detail konfigurasi dari pemberi rekomendasi.

Tipe: Objek RecommenderConfig

Wajib: Tidak

status

Status pemberi rekomendasi. Seorang pemberi rekomendasi dapat berada di salah satu negara bagian berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- STOP PENDING > STOP IN_PROGRESS > TIDAK AKTIF> MULAI TERTUNDA > MULAI IN_PROGRESS > AKTIF
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

RecommenderUpdateSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti pembaruan pemberi rekomendasi. Untuk daftar lengkap, hubungi DescribeRecommenderAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam format Unix) bahwa pembaruan pemberi rekomendasi dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika pembaruan pemberi rekomendasi gagal, alasan di balik kegagalan tersebut.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa pembaruan pemberi rekomendasi terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

recommenderConfig

Detail konfigurasi pembaruan pemberi rekomendasi.

Tipe: Objek RecommenderConfig

Wajib: Tidak

status

Status pembaruan pemberi rekomendasi. Pembaruan pemberi rekomendasi dapat berada di salah satu status berikut:

BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

S3DataConfig

Layanan: Amazon Personalize

Detail konfigurasi bucket input atau output Amazon S3.

Daftar Isi

path

Jalur file bucket Amazon S3.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `(s3|http|https)://.+`

Wajib: Ya

kmsKeyArn

Nama Sumber Daya Amazon (ARN) dari kunci AWS Key Management Service (KMS) yang digunakan Amazon Personalize untuk mengenkripsi atau mendekripsi file input dan output.

Tipe: String

Batasan Panjang: Panjang maksimum 2048.

Pola: `arn:aws.*:kms:.*:[0-9]{12}:key/.*`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Solution

Layanan: Amazon Personalize

> **Important**
>
> Secara default, semua solusi baru menggunakan pelatihan otomatis. Dengan pelatihan otomatis, Anda dikenakan biaya pelatihan saat solusi Anda aktif. Untuk menghindari biaya yang tidak perlu, ketika Anda selesai Anda dapat memperbarui solusi untuk mematikan pelatihan otomatis. Untuk informasi tentang biaya pelatihan, lihat harga Amazon Personalisasi.

Objek yang memberikan informasi tentang solusi. Solusi mencakup resep khusus, parameter yang disesuaikan, dan model terlatih (Versi Solusi) yang digunakan Amazon Personalize untuk menghasilkan rekomendasi.

Setelah Anda membuat solusi, Anda tidak dapat mengubah konfigurasinya. Jika Anda perlu membuat perubahan, Anda dapat mengkloning solusi dengan konsol Amazon Personalize atau membuat yang baru.

Daftar Isi

autoMLResult

Kapan `performAutoML` benar, tentukan resep terbaik yang ditemukan.

Tipe: Objek AutoMLResult

Wajib: Tidak

creationDateTime

Tanggal dan waktu pembuatan (dalam waktu Unix) dari solusi.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup kumpulan data yang menyediakan data pelatihan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

eventType

Jenis acara (misalnya, 'klik' atau 'suka') yang digunakan untuk melatih model. Jika tidak `eventType` disediakan, Amazon Personalize menggunakan semua interaksi untuk pelatihan dengan bobot yang sama terlepas dari jenisnya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa solusi terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

latestSolutionUpdate

Memberikan ringkasan pembaruan terbaru untuk solusi.

Tipe: Objek SolutionUpdateSummary

Wajib: Tidak

latestSolutionVersion

Menjelaskan versi terbaru dari solusi, termasuk status dan ARN.

Tipe: Objek SolutionVersionSummary

Wajib: Tidak

name

Nama solusinya.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

performAutoML

> **Important**
>
> Kami tidak menyarankan untuk mengaktifkan pembelajaran mesin otomatis. Sebagai gantinya, cocokkan kasus penggunaan Anda dengan resep Amazon Personalize yang tersedia. Untuk informasi selengkapnya, lihat Menentukan kasus penggunaan Anda.

Jika benar, Amazon Personalize melakukan pencarian untuk resep USER_PERSONALIZATION terbaik dari daftar yang ditentukan dalam konfigurasi solusi (tidak `recipeArn` boleh ditentukan). Jika false (default), Amazon Personalize digunakan `recipeArn` untuk pelatihan.

Tipe: Boolean

Wajib: Tidak

performAutoTraining

Menentukan apakah solusi secara otomatis membuat versi solusi. Defaultnya adalah `True` dan solusinya secara otomatis membuat versi solusi baru setiap 7 hari.

Untuk informasi selengkapnya tentang auto training, lihat Membuat dan mengonfigurasi solusi.

Tipe: Boolean

Wajib: Tidak

performHPO

Apakah akan melakukan optimasi hyperparameter (HPO) pada resep yang dipilih. Defaultnya adalah `false`.

Tipe: Boolean

Wajib: Tidak

recipeArn

ARN dari resep yang digunakan untuk membuat solusi. Ini diperlukan ketika `performAutoML` salah.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

solutionArn

ARN dari solusinya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

solutionConfig

Menjelaskan properti konfigurasi untuk solusi.

Tipe: Objek SolutionConfig

Wajib: Tidak

status

Status solusinya.

Solusi dapat berada di salah satu negara berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SolutionConfig

Layanan: Amazon Personalize

Menjelaskan properti konfigurasi untuk solusi.

Daftar Isi

algorithmHyperParameters

Daftar algoritma hyperparameters dan nilainya.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

autoMLConfig

MLConfigObjek Auto yang berisi daftar resep untuk mencari ketika AutoML dilakukan.

Tipe: Objek AutoMLConfig

Wajib: Tidak

autoTrainingConfig

Menentukan konfigurasi pelatihan otomatis untuk digunakan.

Tipe: Objek AutoTrainingConfig

Wajib: Tidak

eventsConfig

Menjelaskan konfigurasi suatu peristiwa, yang mencakup daftar parameter peristiwa. Anda dapat menentukan hingga 10 parameter acara. Acara digunakan dalam pembuatan solusi.

Tipe: Objek EventsConfig

Wajib: Tidak

eventValueThreshold

Hanya peristiwa dengan nilai lebih besar dari atau sama dengan ambang batas ini yang digunakan untuk melatih model.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

featureTransformationParameters

Daftar parameter transformasi fitur.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

hpoConfig

Menjelaskan properti untuk optimasi hyperparameter (HPO).

Tipe: Objek HPOConfig

Wajib: Tidak

optimizationObjective

Menjelaskan tujuan tambahan untuk solusi, seperti memaksimalkan menit streaming atau meningkatkan pendapatan. Untuk informasi selengkapnya lihat Mengoptimalkan solusi.

Tipe: Objek OptimizationObjective

Wajib: Tidak

trainingDataConfig

Menentukan konfigurasi data pelatihan yang akan digunakan saat membuat versi solusi kustom (model terlatih).

Tipe: Objek TrainingDataConfig

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SolutionSummary

Layanan: Amazon Personalize

Memberikan ringkasan sifat-sifat solusi. Untuk daftar lengkap, hubungi DescribeSolutionAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa solusi dibuat.

Tipe: Timestamp

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa solusi terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama solusinya.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

recipeArn

Nama Sumber Daya Amazon (ARN) dari resep yang digunakan oleh solusi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

solutionArn

Nama Sumber Daya Amazon (ARN) dari solusinya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status solusinya.

Solusi dapat berada di salah satu negara berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL
- HAPUS TERTUNDA > HAPUS IN_PROGRESS

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SolutionUpdateConfig

Layanan: Amazon Personalize

Detail konfigurasi pembaruan solusi.

Daftar Isi

autoTrainingConfig

Konfigurasi pelatihan otomatis untuk digunakan kapan `performAutoTraining` benar.

Tipe: Objek AutoTrainingConfig

Wajib: Tidak

eventsConfig

Menjelaskan konfigurasi suatu peristiwa, yang mencakup daftar parameter peristiwa. Anda dapat menentukan hingga 10 parameter acara. Acara digunakan dalam pembuatan solusi.

Tipe: Objek EventsConfig

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SolutionUpdateSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti pembaruan solusi. Untuk daftar lengkap, hubungi DescribeSolutionAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam format Unix) bahwa pembaruan solusi dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika pembaruan solusi gagal, alasan di balik kegagalan.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa pembaruan solusi terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

performAutoTraining

Apakah solusi secara otomatis membuat versi solusi.

Tipe: Boolean

Wajib: Tidak

solutionUpdateConfig

Rincian konfigurasi solusi.

Tipe: Objek SolutionUpdateConfig

Wajib: Tidak

status

Status pembaruan solusi. Pembaruan solusi dapat berada di salah satu status berikut:

BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SolutionVersion

Layanan: Amazon Personalize

Objek yang menyediakan informasi tentang versi tertentu dari Solusi dalam grup kumpulan data Kustom.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa versi solusi ini dibuat.

Tipe: Timestamp

Wajib: Tidak

datasetGroupArn

Nama Sumber Daya Amazon (ARN) dari grup dataset yang menyediakan data pelatihan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

eventType

Jenis acara (misalnya, 'klik' atau 'suka') yang digunakan untuk melatih model.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

failureReason

Jika pelatihan versi solusi gagal, alasan kegagalan.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa solusi terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

name

Nama versi solusinya.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

performAutoML

Jika benar, Amazon Personalisasi mencari resep paling optimal sesuai dengan konfigurasi solusi. Saat false (default), Amazon Personalize menggunakan. `recipeArn`

Tipe: Boolean

Wajib: Tidak

performHPO

Apakah akan melakukan optimasi hyperparameter (HPO) pada resep yang dipilih. Defaultnya adalah `false`.

Tipe: Boolean

Wajib: Tidak

recipeArn

ARN dari resep yang digunakan dalam larutan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

solutionArn

ARN dari solusinya.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

solutionConfig

Menjelaskan properti konfigurasi untuk solusi.

Tipe: Objek SolutionConfig

Wajib: Tidak

solutionVersionArn

ARN dari versi solusi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status versi solusi.

Versi solusi dapat berada di salah satu status berikut:

- BUAT TERTUNDA
- BUAT IN_PROGRESS
- AKTIF
- BUAT GAGAL
- BUAT BERHENTI

- BUAT BERHENTI

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

trainingHours

Waktu yang digunakan untuk melatih model. Anda ditagih untuk waktu yang dibutuhkan untuk melatih model. Bidang ini hanya terlihat setelah Amazon Personalize berhasil melatih model.

Tipe: Ganda

Rentang yang Valid: Nilai minimum 0.

Wajib: Tidak

trainingMode

Ruang lingkup pelatihan yang akan dilakukan saat membuat versi solusi. `FULL`Pelatihan mempertimbangkan semua data dalam grup dataset Anda. Suatu `UPDATE` proses hanya data yang telah berubah sejak pelatihan terbaru. Hanya versi solusi yang dibuat dengan resep User-Personalization yang dapat digunakan. `UPDATE`

Tipe: String

Nilai yang Valid: `FULL | UPDATE | AUTOTRAIN`

Wajib: Tidak

trainingType

Apakah versi solusi dibuat secara otomatis atau manual.

Tipe: String

Nilai yang Valid: `AUTOMATIC | MANUAL`

Wajib: Tidak

tunedHPOParams

Jika optimasi hyperparameter dilakukan, berisi nilai hyperparameter dari model berkinerja terbaik.

Tipe: Objek TunedHPOParams

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

SolutionVersionSummary

Layanan: Amazon Personalize

Memberikan ringkasan properti versi solusi. Untuk daftar lengkap, hubungi DescribeSolutionVersionAPI.

Daftar Isi

creationDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa versi solusi ini dibuat.

Tipe: Timestamp

Wajib: Tidak

failureReason

Jika versi solusi gagal, alasan di balik kegagalan.

Tipe: String

Wajib: Tidak

lastUpdatedDateTime

Tanggal dan waktu (dalam waktu Unix) bahwa versi solusi terakhir diperbarui.

Tipe: Timestamp

Wajib: Tidak

solutionVersionArn

Nama Sumber Daya Amazon (ARN) dari versi solusi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

status

Status versi solusi.

Versi solusi dapat berada di salah satu status berikut:

- BUAT TERTUNDA > BUAT IN_PROGRESS> AKTIF -atau- BUAT GAGAL

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

trainingMode

Ruang lingkup pelatihan yang akan dilakukan saat membuat versi solusi. `FULL`Pelatihan mempertimbangkan semua data dalam grup dataset Anda. Suatu `UPDATE` proses hanya data yang telah berubah sejak pelatihan terbaru. Hanya versi solusi yang dibuat dengan resep User-Personalization yang dapat digunakan. `UPDATE`

Tipe: String

Nilai yang Valid: `FULL | UPDATE | AUTOTRAIN`

Wajib: Tidak

trainingType

Apakah versi solusi dibuat secara otomatis atau manual.

Tipe: String

Nilai yang Valid: `AUTOMATIC | MANUAL`

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Tag

Layanan: Amazon Personalize

Metadata opsional yang Anda terapkan pada sumber daya untuk membantu Anda mengkategorikan dan mengaturnya. Setiap tag terdiri dari kunci dan nilai opsional, yang keduanya Anda tentukan. Untuk informasi selengkapnya, lihat Menandai sumber daya Amazon Personalisasi.

Daftar Isi

tagKey

Salah satu bagian dari pasangan kunci-nilai yang membentuk tag. Kunci adalah label umum yang bertindak seperti kategori untuk nilai tanda yang lebih spesifik.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 128.

Pola: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Wajib: Ya

tagValue

Bagian opsional dari pasangan kunci-nilai yang membentuk tag. Nilai bertindak sebagai deskriptor dalam kategori tanda (kunci).

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `^([\p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Diperlukan: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

ThemeGenerationConfig

Layanan: Amazon Personalize

Detail konfigurasi untuk menghasilkan tema dengan pekerjaan inferensi batch.

Daftar Isi

fieldsForThemeGeneration

Bidang yang digunakan untuk menghasilkan tema deskriptif untuk pekerjaan inferensi batch.

Tipe: Objek FieldsForThemeGeneration

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TrainingDataConfig

Layanan: Amazon Personalize

Konfigurasi data pelatihan yang akan digunakan saat membuat pemberi rekomendasi domain atau versi solusi khusus (model terlatih).

Daftar Isi

excludedDatasetColumns

Menentukan kolom untuk mengecualikan dari pelatihan. Setiap kunci adalah tipe dataset, dan setiap nilai adalah daftar kolom. Kecualikan kolom untuk mengontrol data apa yang digunakan Amazon Personalize untuk menghasilkan rekomendasi.

Misalnya, Anda mungkin memiliki kolom yang ingin Anda gunakan hanya untuk memfilter rekomendasi. Anda dapat mengecualikan kolom ini dari pelatihan dan Amazon Personalize mempertimbangkannya hanya saat memfilter.

Tipe: Peta string ke array string

Entri Peta: Jumlah maksimum 3 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Anggota Array: Jumlah maksimum 50 item.

Kendala Panjang: Panjang maksimum 150.

Pola: `[A-Za-z_][A-Za-z\d_]*`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

TunedHPOParams

Layanan: Amazon Personalize

Jika optimasi hyperparameter (HPO) dilakukan, berisi nilai hyperparameter dari model berkinerja terbaik.

Daftar Isi

algorithmHyperParameters

Daftar nilai hyperparameter dari model berkinerja terbaik.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 100 item.

Kendala Panjang Kunci: Panjang maksimum 256.

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Kejadian Amazon Personalize

Tipe data berikut didukung oleh Amazon Personalize Events:

- Action
- ActionInteraction
- Event
- Item

- MetricAttribution
- User

Action

Layanan: Amazon Personalize Events

Merupakan metadata tindakan yang ditambahkan ke kumpulan data Action menggunakan API. `PutActions` Untuk informasi selengkapnya, lihat Mengimpor tindakan secara individual.

Daftar Isi

actionId

ID yang terkait dengan tindakan.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

properties

Peta string metadata khusus tindakan. Setiap elemen dalam peta terdiri dari pasangan kunci-nilai. Misalnya, `{"value": "100"}`.

Kunci menggunakan nama kasus unta yang cocok dengan bidang dalam skema untuk kumpulan data Tindakan. Dalam contoh sebelumnya, `value` cocok dengan bidang 'VALUE' yang ditentukan dalam skema Tindakan. Untuk data string kategoris, untuk menyertakan beberapa kategori untuk satu tindakan, pisahkan setiap kategori dengan pemisah pipa (). `|` Misalnya, `\"Deluxe|Premium\"`.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 32000.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

ActionInteraction

Layanan: Amazon Personalize Events

Merupakan peristiwa interaksi tindakan yang dikirim menggunakan `PutActionInteractions` API.

Daftar Isi

actionId

ID tindakan yang berinteraksi dengan pengguna. Ini sesuai dengan `ACTION_ID` bidang skema interaksi Tindakan.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

eventType

Jenis peristiwa interaksi tindakan. Anda dapat menentukan`Viewed`,`Taken`, dan jenis `Not Taken` acara. Untuk informasi selengkapnya tentang data tipe peristiwa interaksi tindakan, lihat Data tipe peristiwa.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

sessionId

ID yang terkait dengan kunjungan pengguna. Aplikasi Anda menghasilkan unik `sessionId` ketika pengguna pertama kali mengunjungi situs web Anda atau menggunakan aplikasi Anda.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

timestamp

Stempel waktu kapan peristiwa interaksi aksi terjadi. Stempel waktu harus dalam format waktu epoch Unix, dalam hitungan detik.

Tipe: Timestamp

Wajib: Ya

eventId

ID yang terkait dengan acara tersebut. Jika ID peristiwa tidak disediakan, Amazon Personalize akan menghasilkan ID unik untuk acara tersebut. ID peristiwa tidak digunakan sebagai input ke model. Amazon Personalize menggunakan ID peristiwa untuk membedakan peristiwa unik. Setiap peristiwa berikutnya setelah yang pertama dengan ID acara yang sama tidak digunakan dalam pelatihan model.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Tidak

impression

Daftar tindakan IDs yang mewakili urutan tindakan yang telah Anda tunjukkan kepada pengguna. Misalnya, `["actionId1", "actionId2", "actionId3"]`. Amazon Personalize tidak menggunakan data tayangan dari peristiwa interaksi tindakan. Sebagai gantinya, rekam beberapa peristiwa untuk setiap tindakan dan gunakan jenis `Viewed` acara.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 25 item.

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Tidak

properties

Peta string data khusus peristiwa yang mungkin Anda pilih untuk direkam. Misalnya, jika pengguna mengambil tindakan, selain ID tindakan, Anda juga dapat mengirimkan jumlah tindakan yang diambil oleh pengguna.

Setiap item di peta terdiri dari pasangan kunci-nilai. Misalnya,

`{"numberOfActions": "12"}`

Kunci menggunakan nama kasus unta yang cocok dengan bidang dalam skema Interaksi aksi. Dalam contoh di atas, `numberOfActions` akan cocok dengan bidang 'NUMBER_OF_ACTIONS' yang didefinisikan dalam skema interaksi Tindakan.

Berikut ini tidak dapat dimasukkan sebagai kata kunci untuk properti (case insensitive).

- userId
- sessionId
- eventType
- timestamp
- RecommendationID
- kesan

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Wajib: Tidak

recommendationId

ID daftar rekomendasi yang berisi tindakan yang berinteraksi dengan pengguna.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 40.

Wajib: Tidak

userId

ID pengguna yang berinteraksi dengan tindakan. Ini sesuai dengan `USER_ID` bidang skema interaksi Tindakan.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Event

Layanan: Amazon Personalize Events

Merupakan informasi peristiwa interaksi item yang dikirim menggunakan `PutEvents` API.

Daftar Isi

eventType

Jenis acara, seperti klik atau unduh. Properti ini sesuai dengan `EVENT_TYPE` bidang skema kumpulan data interaksi Item Anda dan bergantung pada jenis peristiwa yang Anda lacak.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

sentAt

Stempel waktu (dalam waktu Unix) di sisi klien saat peristiwa terjadi.

Tipe: Timestamp

Wajib: Ya

eventId

ID yang terkait dengan acara tersebut. Jika ID peristiwa tidak disediakan, Amazon Personalize akan menghasilkan ID unik untuk acara tersebut. ID peristiwa tidak digunakan sebagai masukan ke model. Amazon Personalize menggunakan ID peristiwa untuk membedakan peristiwa unik. Setiap peristiwa berikutnya setelah yang pertama dengan ID acara yang sama tidak digunakan dalam pelatihan model.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Tidak

eventValue

Nilai peristiwa yang sesuai dengan `EVENT_VALUE` bidang skema interaksi Item.

tipe: Mengambang

Wajib: Tidak

impression

Daftar item IDs yang mewakili urutan item yang telah Anda tunjukkan kepada pengguna. Misalnya, `["itemId1", "itemId2", "itemId3"]`. Berikan daftar item untuk merekam data tayangan secara manual untuk suatu peristiwa. Untuk informasi selengkapnya tentang merekam data tayangan, lihat Merekam data tayangan.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 25 item.

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Tidak

itemId

Kunci ID item yang sesuai dengan `ITEM_ID` bidang skema kumpulan data interaksi Item.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Tidak

metricAttribution

Berisi informasi tentang atribusi metrik yang terkait dengan suatu peristiwa. Untuk informasi selengkapnya tentang atribusi metrik, lihat Mengukur dampak rekomendasi.

Tipe: Objek MetricAttribution

Wajib: Tidak

properties

Peta string data khusus peristiwa yang mungkin Anda pilih untuk direkam. Misalnya, jika pengguna menilai film di situs Anda, selain ID film (`itemId`) dan rating (`eventValue`), Anda juga dapat mengirimkan jumlah peringkat film yang dibuat oleh pengguna.

Setiap item di peta terdiri dari pasangan kunci-nilai. Misalnya,

```
{"numberOfRatings": "12"}
```

Kunci menggunakan nama kasus unta yang cocok dengan bidang dalam skema kumpulan data interaksi Item. Dalam contoh di atas, `numberOfRatings` akan cocok dengan bidang 'NUMBER_OF_RATINGS' yang ditentukan dalam skema kumpulan data interaksi Item.

Berikut ini tidak dapat dimasukkan sebagai kata kunci untuk properti (case insensitive).

- userId
- sessionId
- eventType
- timestamp
- RecommendationID
- kesan

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Wajib: Tidak

recommendationId

ID daftar rekomendasi yang berisi item yang berinteraksi dengan pengguna. Berikan `recommendationId` agar Amazon Personalisasi secara implisit merekam rekomendasi yang Anda tunjukkan kepada pengguna sebagai data tayangan. Atau berikan `recommendationId` jika Anda menggunakan atribusi metrik untuk mengukur dampak rekomendasi.

Untuk informasi selengkapnya tentang merekam data tayangan, lihat Merekam data tayangan. Untuk informasi selengkapnya tentang membuat atribusi metrik, lihat Mengukur dampak rekomendasi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 40.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Item

Layanan: Amazon Personalize Events

Merupakan metadata item yang ditambahkan ke kumpulan data Item menggunakan API. `PutItems` Untuk informasi selengkapnya lihat Mengimpor item satu per satu.

Daftar Isi

itemId

ID yang terkait dengan item.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

properties

Peta string metadata khusus item. Setiap elemen dalam peta terdiri dari pasangan kunci-nilai. Misalnya, `{"numberOfRatings": "12"}`.

Kunci menggunakan nama kasus unta yang cocok dengan bidang dalam skema untuk dataset Item. Pada contoh sebelumnya, kolom `numberOfRatings` cocok dengan 'NUMBER_OF_RATINGS' yang ditentukan dalam skema Item. Untuk data string kategoris, untuk menyertakan beberapa kategori untuk satu item, pisahkan setiap kategori dengan pemisah pipa (). `|` Misalnya, `\"Horror|Action\"`.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 32000.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

MetricAttribution

Layanan: Amazon Personalize Events

Berisi informasi tentang atribusi metrik yang terkait dengan suatu peristiwa. Untuk informasi selengkapnya tentang atribusi metrik, lihat Mengukur dampak rekomendasi.

Daftar Isi

eventAttributionSource

Sumber acara, seperti pihak ketiga.

Tipe: String

Batasan Panjang: Panjang maksimum 1024.

Pola: `^[\x20-\x7E]*[\x21-\x7E]+[\x20-\x7E]*$`

Diperlukan: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

User

Layanan: Amazon Personalize Events

Merupakan metadata pengguna yang ditambahkan ke kumpulan data Pengguna menggunakan API. `PutUsers` Untuk informasi selengkapnya lihat Mengimpor pengguna satu per satu.

Daftar Isi

userId

ID yang terkait dengan pengguna.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Wajib: Ya

properties

Peta string metadata khusus pengguna. Setiap elemen dalam peta terdiri dari pasangan kunci-nilai. Misalnya, `{"numberOfVideosWatched": "45"}`.

Kunci menggunakan nama kasus unta yang cocok dengan bidang dalam skema untuk kumpulan data Pengguna. Pada contoh sebelumnya, kolom `numberOfVideosWatched` cocok dengan 'NUMBER_OF_VIDEOS_WATCHED' yang ditentukan dalam skema Pengguna. Untuk data string kategoris, untuk menyertakan beberapa kategori untuk satu pengguna, pisahkan setiap kategori dengan pemisah pipa (). `|` Misalnya, `\"Member|Frequent shopper\"`.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 24000.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

Waktu Aktif Amazon Personalize

Tipe data berikut ini didukung oleh Amazon Personalize Runtime:

- PredictedAction
- PredictedItem
- Promotion

PredictedAction

Layanan: Amazon Personalize Runtime

Objek yang mengidentifikasi suatu tindakan.

GetActionRecommendationsAPI mengembalikan daftar `PredictedAction` s.

Daftar Isi

actionId

ID dari tindakan yang direkomendasikan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

score

Skor tindakan yang direkomendasikan. Untuk informasi tentang skor aksi, lihat Cara kerja penilaian rekomendasi tindakan.

Tipe: Ganda

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

PredictedItem

Layanan: Amazon Personalize Runtime

Objek yang mengidentifikasi item.

GetRecommendationsDan GetPersonalizedRanking APIs mengembalikan daftar `PredictedItem` s.

Daftar Isi

itemId

ID item yang direkomendasikan.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

metadata

Metadata tentang item dari dataset Item Anda.

Tipe: Peta string ke string

Kendala Panjang Kunci: Panjang maksimum 150.

Kendala Panjang Nilai: Panjang maksimum 20000.

Wajib: Tidak

promotionName

Nama promosi yang termasuk item yang diprediksi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

reason

Jika Anda menggunakan User-Personalization-v 2, daftar alasan mengapa item tersebut dimasukkan dalam rekomendasi. Alasan yang mungkin termasuk yang berikut:

- Item promosi - Menunjukkan item tersebut disertakan sebagai bagian dari promosi yang Anda terapkan dalam permintaan rekomendasi Anda.
- Eksplorasi - Menunjukkan item itu disertakan dengan eksplorasi. Dengan eksplorasi, rekomendasi mencakup item dengan data interaksi yang lebih sedikit atau relevansi bagi pengguna. Untuk informasi lebih lanjut tentang eksplorasi, lihat Eksplorasi.
- Item populer - Menunjukkan item dimasukkan sebagai item populer placeholder. Jika Anda menggunakan filter, tergantung pada berapa banyak rekomendasi yang dihapus filter, Amazon Personalize dapat menambahkan item placeholder untuk memenuhi permintaan rekomendasi Anda. `numResults` Item ini adalah item populer, berdasarkan data interaksi, yang memenuhi kriteria filter Anda. Mereka tidak memiliki skor relevansi bagi pengguna.

Tipe: Array string

Batasan Panjang: Panjang maksimum 256.

Wajib: Tidak

score

Representasi numerik dari kepastian model bahwa item tersebut akan menjadi pilihan pengguna berikutnya. Untuk informasi lebih lanjut tentang logika penilaian, lihatSkor rekomendasi.

Tipe: Ganda

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Promotion

Layanan: Amazon Personalize Runtime

Berisi informasi tentang promosi. Promosi mendefinisikan aturan bisnis tambahan yang berlaku untuk subset item yang direkomendasikan yang dapat dikonfigurasi.

Daftar Isi

filterArn

Nama Sumber Daya Amazon (ARN) dari filter yang digunakan oleh promosi. Filter ini mendefinisikan kriteria untuk item yang dipromosikan. Untuk informasi selengkapnya, lihat Filter promosi.

Tipe: String

Batasan Panjang: Panjang maksimum 256.

Pola: `arn:([a-z\d-]+):personalize:.*:.*:.+`

Wajib: Tidak

filterValues

Nilai yang digunakan saat mempromosikan item. Untuk setiap parameter placeholder dalam ekspresi filter promosi Anda, berikan nama parameter (dalam kasus yang cocok) sebagai kunci dan nilai filter sebagai nilai yang sesuai. Pisahkan beberapa nilai untuk satu parameter dengan koma.

Untuk ekspresi filter yang menggunakan `INCLUDE` elemen untuk menyertakan item, Anda harus memberikan nilai untuk semua parameter yang ditentukan dalam ekspresi. Untuk filter dengan ekspresi yang menggunakan `EXCLUDE` elemen untuk mengecualikan item, Anda dapat menghilangkan. `filter-values` Dalam kasus ini, Amazon Personalize tidak menggunakan bagian ekspresi tersebut untuk memfilter rekomendasi.

Untuk informasi selengkapnya tentang membuat filter, lihat Memfilter rekomendasi dan segmen pengguna.

Tipe: Peta string ke string

Entri Peta: Jumlah maksimum 25 item.

Kendala Panjang Kunci: Panjang maksimum 50.

Pola Kunci: `[A-Za-z0-9_]+`

Kendala Panjang Nilai: Panjang maksimum 1000.

Wajib: Tidak

name

Nama promosi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 63.

Pola: `^[a-zA-Z0-9][a-zA-Z0-9\-_]*`

Wajib: Tidak

percentPromotedItems

Persentase item yang direkomendasikan untuk menerapkan promosi.

Tipe: Bilangan Bulat

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 100.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan API ini di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

Kesalahan Umum

Bagian ini mencantumkan kesalahan yang umum terjadi pada tindakan API dari semua AWS layanan. Untuk kesalahan khusus pada tindakan API untuk layanan ini, lihat topik untuk tindakan API tersebut.

AccessDeniedException

Anda tidak memiliki akses yang memadai untuk melakukan tindakan ini.

Kode Status HTTP: 400

IncompleteSignature

Tanda tangan permintaan tidak sesuai dengan AWS standar.

Kode Status HTTP: 400

InternalFailure

Pemrosesan permintaan telah gagal karena kesalahan yang tidak diketahui, pengecualian atau kegagalan.

Kode Status HTTP: 500

InvalidAction

Tindakan atau operasi yang diminta tidak valid. Verifikasi bahwa tindakan diketik dengan benar.

Kode Status HTTP: 400

InvalidClientTokenId

Sertifikat X.509 atau ID kunci AWS akses yang disediakan tidak ada dalam catatan kami.

Kode Status HTTP: 403

NotAuthorized

Anda tidak memiliki izin untuk melakukan tindakan ini.

Kode Status HTTP: 400

OptInRequired

ID kunci AWS akses memerlukan langganan untuk layanan ini.

Kode Status HTTP: 403

RequestExpired

Permintaan mencapai layanan lebih dari 15 menit setelah cap tanggal pada permintaan atau lebih dari 15 menit setelah tanggal kedaluwarsa permintaan (seperti untuk pra-ditandatangani URLs), atau cap tanggal pada permintaan lebih dari 15 menit di masa depan.

Kode Status HTTP: 400

ServiceUnavailable

Permintaan telah gagal karena kegagalan sementara server.

Kode Status HTTP: 503

ThrottlingException

Permintaan ditolak karena throttling permintaan.

Kode Status HTTP: 400

ValidationError

Input gagal memenuhi kendala yang ditentukan oleh layanan. AWS

Kode Status HTTP: 400

Parameter Umum

Daftar berikut berisi parameter yang digunakan semua tindakan untuk menandatangani permintaan Tanda Tangan Versi 4 dengan string kueri. Setiap parameter khusus tindakan tercantum dalam topik untuk tindakan tersebut. Untuk informasi selengkapnya tentang Tanda Tangan Versi 4, lihat Menandatangani permintaan AWS API di Panduan Pengguna IAM.

Action

Tindakan yang harus dilakukan.

Tipe: string

Diperlukan: Ya

Version

Versi API tempat permintaan ditulis, dinyatakan dalam format YYYY-MM-DD.

Jenis: string

Diperlukan: Ya

X-Amz-Algorithm

Algoritme hash yang Anda gunakan untuk membuat tanda tangan permintaan.

Syarat: Tentukan parameter ini ketika Anda menyertakan informasi autentikasi dalam string kueri alih-alih di header otorisasi HTTP.

Tipe: string

Nilai yang Valid: `AWS4-HMAC-SHA256`

Diperlukan: Kondisional

X-Amz-Credential

Nilai lingkup kredensial, yang merupakan string yang menyertakan access key Anda, tanggal, wilayah yang Anda targetkan, layanan yang Anda minta, dan string penghentian ("aws4_request"). Nilai dinyatakan dalam format berikut: access_key/HHBBTTTT/wilayah/layanan/aws4_request.

Untuk informasi selengkapnya, lihat Membuat permintaan AWS API yang ditandatangani di Panduan Pengguna IAM.

Syarat: Tentukan parameter ini ketika Anda menyertakan informasi autentikasi dalam string kueri alih-alih di header otorisasi HTTP.

Tipe: string

Wajib: Bersyarat

X-Amz-Date

Tanggal yang digunakan untuk membuat tanda tangan. Format harus berupa format dasar ISO 8601 (YYYYMMDD'T'HMMSS'Z'). Misalnya, waktu tanggal berikut adalah X-Amz-Date nilai yang valid:`20120325T120000Z`.

Syarat: X-Amz-Date bersifat opsional untuk semua permintaan; ini dapat digunakan untuk menimpa tanggal yang digunakan untuk menandatangani permintaan. Jika header Tanggal ditentukan dalam format dasar ISO 8601, tidak X-Amz-Date diperlukan. Ketika X-Amz-Date digunakan, itu selalu mengesampingkan nilai header Tanggal. Untuk informasi selengkapnya, lihat Elemen tanda tangan permintaan AWS API di Panduan Pengguna IAM.

Jenis: string

Wajib: Bersyarat

X-Amz-Security-Token

Token keamanan sementara yang diperoleh melalui panggilan ke AWS Security Token Service (AWS STS). Untuk daftar layanan yang mendukung kredensi keamanan sementara AWS STS, lihat layanan Layanan AWS yang berfungsi dengan IAM di Panduan Pengguna IAM.

Kondisi: Jika Anda menggunakan kredensi keamanan sementara dari AWS STS, Anda harus menyertakan token keamanan.

Jenis: string

Wajib: Bersyarat

X-Amz-Signature

Menentukan tanda tangan yang dikodekan oleh hex yang dihitung dari string to sign dan kunci penandatanganan turunan.

Syarat: Tentukan parameter ini ketika Anda menyertakan informasi autentikasi dalam string kueri alih-alih di header otorisasi HTTP.

Tipe: string

Wajib: Bersyarat

X-Amz-SignedHeaders

Menentukan semua header HTTP yang disertakan sebagai bagian dari permintaan kanonik. Untuk informasi selengkapnya tentang menentukan header yang ditandatangani, lihat Membuat permintaan AWS API yang ditandatangani di Panduan Pengguna IAM.

Syarat: Tentukan parameter ini ketika Anda menyertakan informasi autentikasi dalam string kueri alih-alih di header otorisasi HTTP.

Tipe: string

Diperlukan: Kondisional

Riwayat dokumen untuk Amazon Personalisasi

Tabel berikut menjelaskan perubahan penting dalam setiap rilis Panduan Pengembang Amazon Personalize. Untuk notifikasi tentang pembaruan dokumentasi ini, Anda dapat berlangganan ke umpan RSS.

Perubahan	Deskripsi	Tanggal
Fitur baru	Amazon Personalize sekarang mendukung solusi konfigurasi untuk memberikan bobot yang berbeda untuk jenis acara yang berbeda. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi dengan konfigurasi peristiwa.	April 11, 2025
Fitur baru	Amazon Personalize sekarang mendukung IPv6 lalu lintas menggunakan titik akhir dualstack. Untuk daftar titik akhir Amazon Personalisasi menurut wilayah, lihat AWS wilayah dan titik akhir di Referensi Umum Amazon Web Services.	Januari 31, 2025
Fitur dokumentasi baru	Dengan data tampilan halaman dan analisis kedalaman, struktur panduan pengembang Amazon Personalize diperbarui untuk meningkatkan visibilitas topik penting. Navigasi diperbarui untuk mengurangi kedalaman keseluruhan dan topik terkait dikonsolidasikan.	September 3, 2024

	Misalnya, Anda sekarang dapat menemukan semua panduan persiapan data dalam topik Mempersiapkan data pelatihan untuk Amazon Personalisasi. Untuk topik yang dapat membantu Anda menavigasi struktur baru, lihat Amazon Personalisasi detail alur kerja. Halaman ini memetakan setiap langkah alur kerja Amazon Personali ze ke topik terkait dalam panduan.	
Fitur baru	Anda sekarang dapat memperbarui solusi kustom Amazon Personalize yang ada untuk mengubah konfigura si pelatihannya. Anda dapat mengaktifkan atau menonakti fkan pelatihan otomatis, dan Anda dapat mengubah frekuensi pelatihan. Untuk informasi selengkapnya, lihat Memperbarui solusi untuk mengubah konfigurasi pelatihan otomatisnya.	Agustus 29, 2024

Fitur dokumentasi baru	Hasil penelusuran personalisasi dari OpenSearch topik dalam panduan pengembang Amazon Personalize direorganisasi OpenSearch oleh Service dan alur kerja open source. OpenSearch Sekarang fitur sub topik baru: Personalisasi hasil dari Amazon OpenSearch Service dengan Amazon Personalisasi dan Personalisasi hasil dari open source Open Search dengan Amazon Personalize.	Agustus 23, 2024
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk menghapus pengguna dan data mereka, termasuk metadata dan data interaksi mereka, dari grup kumpulan data. Untuk informasi selengkapnya, lihat Menghapus pengguna.	2 Mei 2024
Fitur baru	Amazon Personalize sekarang dapat mempertimbangkan hingga 5 juta item dengan pelatihan yang lebih cepat, latensi rekomendasi yang lebih rendah, dan rekomendasi yang lebih relevan. Dua resep kustom baru menggunakan pelatihan yang disempurnakan ini, User-personalization-v2 dan Personalized-ranking-v2.	1 Mei 2024

Fitur baru	Semua Amazon Personalisasi solusi kustom baru sekarang menggunakan pelatihan otomatis. Untuk informasi selengkapnya tentang mengonfigurasi pelatihan otomatis, lihat Mengonfigurasi pelatihan otomatis.	April 19, 2024
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk merekomendasikan tindakan terbaik berikutnya untuk pengguna berdasark an perilaku mereka. Untuk informasi lebih lanjut, lihat Next-Best-Action resep.	26 November 2023
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk memasukkan tema deskriptif dalam rekomenda si batch dengan bantuan AI generatif. Untuk informasi tentang membuat rekomenda si batch dengan tema, lihat Rekomendasi Batch dengan tema. Untuk informasi tentang penggunaan Amazon Personalize dengan AI generatif, lihat Amazon Personalize dan Generative AI.	26 November 2023

Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk menyertakan metadata dari kumpulan data Item dalam rekomendasi. Untuk informasi tentang mengaktifkan metadata untuk kampanye, lihat Metadata item dalam rekomendasi. Untuk informasi tentang mengaktifkan metadata untuk pemberi rekomendasi, lihat Mengaktifkan metadata dalam rekomendasi untuk pemberi rekomendasi.	26 November 2023
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk mempersonalisasi hasil pencarian dari Layanan. OpenSearch Untuk informasi selengkapnya, lihat Mempersonalisasi hasil penelusuran dari OpenSearch.	16 Oktober 2023

Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk mengimpor hingga 100 kolom metadata untuk kumpulan data Item (naik dari 50) dan 25 kolom metadata untuk kumpulan data Pengguna (naik dari 5). Untuk informasi selengkapnya tentang batas Personalisasi Amazon, lihat Amazon Personalisasi titik akhir dan kuota.	5 September 2023
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk mempersonalisasi hasil pencarian dari. OpenSearch Untuk informasi selengkapnya, lihat Mempersonalisasi hasil penelusuran dari OpenSearch (dikelola sendiri).	25 Juli 2023
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk mengganti skema dataset dengan yang baru atau yang sudah ada. Untuk informasi selengkapnya, lihat Mengganti skema kumpulan data.	13 Juli 2023

Fitur baru	Jika Anda menggunakan resep Personalisasi Pengguna atau Personalisasi Peringkat , Amazon Personalisasi pekerjaan inferensi batch sekarang dapat menggunakan data yang Anda impor secara bertahap tanpa pelatihan ulang. Untuk selengkapnya, lihat Mendapatkan rekomendasi batch.	Juni 30, 2023
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk memfilter item berdasarkan item yang Anda tentukan dalam permintaan Anda untuk rekomendasi item terkait. Untuk informasi tentang filter, lihat Memfilter rekomendasi dan segmen pengguna.	Juni 21, 2023
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk memfilter item berdasarkan item yang Anda tentukan dalam permintaan Anda untuk rekomendasi item terkait. Untuk informasi tentang filter, lihat Memfilter rekomendasi dan segmen pengguna.	Juni 21, 2023

Fitur baru	Amazon Personalize sekarang mendukung koneksi pribadi antara virtual private cloud (VPC) dan Amazon Personalize dengan antarmuka Amazon VPC endpoint. Untuk informasi selengkapnya, lihat Amazon Personalisasi dan antarmuka titik akhir VPC ().AWS PrivateLink	12 Juni 2023
Fitur baru	Amazon Personalize sekarang mendukung konfigurasi kolom yang digunakan saat pelatihan saat Anda membuat pemberi rekomendasi atau solusi khusus. Untuk informasi tentang mengonfigurasi kolom saat membuat pemberi rekomendasi, lihat Membuat pemberi rekomendasi. Untuk informasi tentang mengonfigurasi kolom saat membuat solusi, lihat Mengonfigurasi kolom yang digunakan saat latihan.	30 Mei 2023
Fitur dokumentasi baru	Panduan pengembang Amazon Personalize sekarang mencakup informasi tentang melakukan A/B pengujian dengan rekomendasi Amazon Personalize. Untuk informasi selengkapnya, lihat Mengukur dampak rekomendasi dengan A/B pengujian.	5 Mei 2023

Fitur baru	Amazon Personalize sekarang mendukung konfigurasi bagaimana popularitas memengaruhi rekomendasi yang dihasilkan oleh resep Similar-Items. Untuk informasi selengkapnya, lihat resep Item Serupa.	April 21, 2023
Fitur baru	Amazon Personalize sekarang mendukung penggunaan Amazon SageMaker AI Data Wrangler untuk mengimpor data dari 40+ sumber ke kumpulan data Amazon Personalize. Untuk informasi selengkapnya, lihat Mengimpor data menggunakan Amazon SageMaker AI Data Wrangler.	April 14, 2023
Fitur dokumentasi baru	Panduan pengembang Amazon Personalize sekarang menyertakan daftar periksa kesiapan baru. Daftar periksa ini membantu Anda mempersiapkan diri untuk menggunakan Amazon Personalize dengan data Anda sendiri. Untuk informasi selengkapnya, lihat Daftar periksa kesiapan.	9 Februari 2023

Fitur baru	Amazon Personalize sekarang mendukung pembuatan wawasan dan statistik untuk data yang Anda impor ke kumpulan data. Untuk informasi selengkapnya, lihat Menganalisis data dalam kumpulan data.	Januari 25, 2023
Fitur baru	Amazon Personalize sekarang mendukung resep Trending-Now baru untuk grup kumpulan data Kustom. Untuk informasi lebih lanjut, lihat resep Trending-Now.	Januari 6, 2023
Fitur baru	Amazon Personalize sekarang mendukung penggunaan tag dalam kebijakan IAM. Untuk informasi selengkapnya, lihat Menggunakan tag dalam kebijakan IAM.	28 Desember 2022
Fitur baru	Amazon Personalize sekarang mendukung penyesuaian jumlah maksimum interaksi yang dipertimbangkan oleh model selama kuota pelatihan . Selain itu, jumlah maksimum pengguna yang dipertimb angkan oleh model selama kuota batas pelatihan tidak lagi berlaku. Untuk informasi selengkapnya, lihat Amazon Personalisasi titik akhir dan kuota.	15 Desember 2022

Fitur baru	Amazon Personalize sekarang mendukung pembuatan atribusi metrik untuk mengukur dampak bisnis dari rekomendasi. Untuk informasi selengkapnya, lihat Mengukur dampak rekomendasi.	17 November 2022
Fitur baru	Amazon Personalisasi kuota untuk jumlah total solusi aktif, kampanye aktif, pemberi rekomendasi, dan filter kini telah meningkat. Masing-masing kuota ini sekarang berlaku per grup kumpulan data daripada per akun. Untuk informasi tentang kuota, lihat Amazon Personalisasi titik akhir dan kuota.	7 September 2022
Fitur baru	Filter Amazon Personalize sekarang mempertimbangkan hingga 100 interaksi per pengguna per jenis acara. Untuk informasi tentang filter, lihat Memfilter rekomendasi dan segmen pengguna.	Agustus 29, 2022
Fitur baru	Amazon Personalize sekarang mendukung kasus penggunaan Trending now baru untuk domain VIDEO_ON_DEMAND. Untuk informasi selengkapnya, lihat kasus penggunaan VIDEO_ON_DEMAND.	17 Agustus 2022

Fitur baru	Amazon Personalize sekarang mendukung mempromosikan item dalam rekomendasi dengan filter promosi terpisah. Untuk informasi tentang mempromosikan item, lihat Mempromosikan item dalam rekomendasi.	12 Agustus 2022
Fitur baru	Amazon Personalize sekarang mendukung penggunaan operator perbandingan dalam ekspresi filter dengan parameter placeholder. Untuk informasi tentang ekspresi filter, lihat Menyaring ekspresi.	12 Agustus 2022
Fitur baru	Amazon Personalize sekarang mendukung pembaruan massal inkremental ke kumpulan data. Anda sekarang dapat menggunakan pekerjaan impor dataset untuk memperbarui dataset tanpa mengganti data yang ada. Untuk informasi selengkapnya, lihat Memperbarui data massal yang ada.	2 Agustus 2022
Fitur baru	Amazon Personalize sekarang dapat menggunakan metadata teks tidak terstruktur dalam berbagai bahasa. Untuk informasi selengkapnya, lihat Metadata teks tidak terstruktur.	6 Juni 2022

Fitur baru	Amazon Personalize recommended sekarang menghasilkan metrik offline. Anda dapat menggunakan metrik ini untuk mengevaluasi kinerja pemberi rekomenda si Anda. Untuk informasi selengkapnya, lihat Mengevalu asi pemberi rekomendasi.	24 Mei 2022
Fitur baru	Amazon Personalize sekarang mendukung kemampuan menghenti kan pemberi rekomendasi dan memulai ulang nanti. Dengan cara ini, Anda dapat menjeda penagihan pemberi rekomendasi dan membayarnya hanya saat Anda menggunakannya. Untuk informasi selengkapnya, lihat Menghentikan dan memulai pemberi rekomendasi.	20 April 2022
Fitur baru	Amazon Personalize sekarang mendukung penggunaan tag untuk mengkategorikan dan mengelola sumber daya Amazon Personalisasi. Untuk informasi selengkapnya, lihat Menandai sumber daya Amazon Personalisasi.	7 April 2022

Fitur baru	Amazon Personalize sekarang mendukung menentuka n sumber daya dengan. AWS CloudFormation Untuk informasi selengkapnya, lihat Menentukan sumber daya dengan AWS CloudFormation.	11 Maret 2022
Fitur dokumentasi baru	Panduan pengemban g Amazon Personalize sekarang menyertakan topik Pemecahan Masalah baru yang memberikan jawaban atas pertanyaan umum dan saran pemecahan masalah untuk pesan kesalahan yang mungkin Anda temui dengan Amazon Personalize. Untuk informasi selengkapnya, lihat Pemecahan Masalah.	Februari 15, 2022
Fitur baru	Amazon Personalize sekarang mendukung pembuatan grup kumpulan data Domain dengan sumber daya yang dioptimalkan untuk domain video sesuai permintaan atau e-commerce. Untuk informasi selengkapnya, lihat Grup kumpulan data domain.	29 November 2021

Fitur baru	Amazon Personalize sekarang mendukung pembuatan segmen pengguna dengan resep USER_SEGM ENTATION baru. Resep USER_SEGMENTATION menghasilkan segmen pengguna berdasarkan data input item. Untuk informasi lebih lanjut, lihat Resep dan resep Item-Affinity. Item-Attr ibute-Affinity	29 November 2021
Fitur baru	Amazon Personalize sekarang mendukung resep rekomenda si RELATED_ITEMS baru Similar-Items. Gunakan resep Item Serupa untuk menghasil kan rekomendasi untuk item serupa berdasarkan data interaksi dan metadata item. Untuk informasi selengkapnya, lihat resep Item Serupa.	5 Oktober 2021
Fitur dokumentasi baru	Panduan pengembang Amazon Personalize sekarang menyertakan tutorial memulai untuk menggunakan Amazon Personalize dengan SDK for Java 2.x. Untuk informasi selengkapnya, lihat Memulai (SDK for Java 2.x).	25 Agustus 2021

Fitur baru	Amazon Personalize sekarang dapat mengekstrak informasi yang bermakna dari metadata teks tidak terstruktur dalam kumpulan data Item. Untuk informasi selengkapnya, lihat Kumpulan data Item.	9 Juni 2021
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk berhenti membuat versi solusi (berhenti melatih model). Untuk informasi selengkapnya, lihat Menghentikan pembuatan versi solusi.	20 Mei 2021
Fitur baru (rilis pratinjau)	Amazon Personalize sekarang dapat mengoptimalkan solusi untuk tujuan selain memaksimalkan relevansi , seperti memaksima lkan pendapatan. Fitur ini dalam rilis pratinjau. Untuk informasi selengkapnya, lihat Mengoptimalkan solusi untuk tujuan tambahan.	18 Mei 2021
Fitur baru	Amazon Personalize sekarang dapat mengekspor catatan dalam kumpulan data Amazon Personalize ke bucket Amazon S3 untuk analisis dan pelacakan. Untuk informasi selengkapnya, lihat Mengekspor kumpulan data.	26 April 2021

Fitur baru	Amazon Personalize sekarang secara otomatis memperbarui model terbaru (versi solusi) yang Anda latih dengan Personalisasi Pengguna setiap dua jam untuk menyertakan data baru. Untuk informasi selengkapnya, lihat resep personalisasi pengguna.	17 November 2020
Fitur baru	Amazon Personalize sekarang dapat memfilter rekomendasi berdasarkan kriteria yang Anda tentukan saat Anda mendapatkan rekomendasi. Untuk informasi selengkapnya, lihat Memfilter rekomendasi.	10 November 2020
Fitur baru	Amazon Personalize sekarang mendukung kemampuan untuk mengimpor pengguna dan item secara bertahap. Untuk informasi selengkapnya, lihat Mengimpor catatan secara bertahap.	2 Oktober 2020
Fitur baru	Amazon Personalize sekarang mendukung resep rekomendasi USER_PERSONALIZATION baru. Fitur USER_PERSONALIZATION termasuk pemodelan data tayangan, eksplorasi item otomatis, dan pemilihan item dingin otomatis. Untuk informasi selengkapnya, lihat resep personalisasi pengguna.	Agustus 5, 2020

Fitur baru	Amazon Personalize sekarang dapat memfilter rekomendasi berdasarkan item dan metadata pengguna menggunakan ekspresi filter khusus. Untuk informasi selengkapnya, lihat Memfilter rekomendasi.	31 Juli 2020
Fitur baru	Amazon Personalize sekarang memungkinkan Anda memfilter hasil berdasarkan item mana yang berintera ksi dengan pengguna. Untuk informasi selengkapnya, lihat Memfilter rekomendasi.	3 Juni 2020
Fitur baru	Amazon Personalize sekarang memperlihatkan skor untuk item yang direkomendasikan. Skor mewakili kepastian model Amazon Personalize bahwa pengguna selanjutnya akan memilih item tertentu. Untuk informasi selengkapnya, lihat Mendapatkan rekomendasi.	3 April 2020
Wilayah Baru	Amazon Personalize menambahkan dukungan untuk Wilayah Asia Pasifik (Seoul). Untuk daftar lengkap AWS Wilayah yang didukung oleh Amazon Personalize, lihat tabel AWS Wilayah atau AWS Wilayah dan titik akhir di Referensi Umum Amazon Web Services.	21 Januari 2020

Fitur baru	Amazon Personalize sekarang bisa mendapatkan rekomendasi berdasarkan metadata kontekstual. Untuk informasi selengkapnya, lihat Mendapatkan rekomendasi.	19 Desember 2019
Daerah baru	Amazon Personalize menambahkan dukungan untuk Wilayah Asia Pasifik (Mumbai), Asia Pasifik (Sydney), dan Kanada (Tengah). Untuk daftar lengkap AWS Wilayah yang didukung oleh Amazon Personalize, lihat tabel AWS Wilayah atau AWS Wilayah dan titik akhir di Referensi Umum Amazon Web Services.	18 Desember 2019
Fitur baru	Amazon Personalize sekarang mendukung alur kerja rekomendasi batch. Untuk informasi selengkapnya, lihat Mendapatkan rekomendasi batch.	14 November 2019
Amazon Personalisasi ketersediaan umum	Amazon Personalize sekarang tersedia untuk penggunaan umum.	10 Juni 2019
Amazon Personalisasi rilis pratinjau	Ini adalah rilis pratinjau pertama dari dokumentasi untuk Amazon Personalize.	28 November, 2018

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.