



Panduan Developer

Amazon Comprehend Medical



Amazon Comprehend Medical: Panduan Developer

Copyright © 2023 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 para pelanggan, atau dengan cara apa pun yang menghina atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan properti dari masing-masing pemilik, yang mungkin berafiliasi, terkait dengan, atau disponsori oleh Amazon, atau tidak.

Table of Contents

Apa itu Amazon Comprehend Medical?	1
Pemberitahuan penting	1
Amazon Comprehend Medical kasus penggunaan	1
Manfaat Amazon Comprehend Medical	2
Kepatuhan HIPAA	3
Mengakses Amazon Comprehend Medical	3
Cara Memulai Amazon Comprehend Medical	4
Cara kerjanya	5
Deteksi entitas sinkron	5
Analisis batch asinkron	5
Menghubungkan Ontologi	6
Menghubungkan ke konsep dalam basis pengetahuan ICD-10-CM tentang kondisi medis	6
Menghubungkan ke konsep di basis pengetahuan RxNorm obat	6
Menghubungkan ke konsep dalam basis pengetahuan SNOMED CT konsep medis	7
Memulai	8
Langkah 1: Siapkan Akun	8
Daftar ke AWS	8
Membuat Pengguna IAM	9
Langkah selanjutnya	10
Langkah 2: Siapkan AWS CLI	10
Langkah selanjutnya	11
Langkah 3: Memulai Menggunakan Konsol	11
Menganalisis teks klinis menggunakan konsol	11
Langkah 4: Memulai Menggunakan API	14
Mendeteksi entitas medis menggunakan AWS Command Line Interface	14
Mendeteksi entitas medis menggunakan AWS SDK for Java	16
Mendeteksi entitas medis menggunakan AWS SDK for Python (Boto)	17
Titik akhir VPC (AWS PrivateLink)	19
Pertimbangan untuk VPC endpoint Amazon Comprehend Medical	19
Membuat VPC endpoint antarmuka untuk Amazon Comprehend Medical	19
Membuat kebijakan VPC endpoint untuk Amazon Comprehend Medical	20
API	21
Mendeteksi entitas (Versi 2)	21
Kategori anatomi	23

Kategori perilaku, lingkungan, dan kesehatan sosial	23
Kategori kondisi medis	24
Kategori obat	25
Kategori informasi kesehatan yang dilindungi	26
Kategori tes, perawatan, dan prosedur	27
Kategori ekspresi waktu	27
Mendeteksi entitas	28
Kategori anatomi	29
Kategori kondisi medis	31
Kategori obat	33
Kategori Informasi Kesehatan Terlindungi	37
Kategori tes, perawatan, dan prosedur	39
Mendeteksi PHI	42
API batch analisis teks	47
Pemberitahuan penting	1
Melakukan analisis batch menggunakan API	47
Melakukan analisis batch menggunakan konsol	48
Kebijakan IAM	48
File keluaran analisis Batch	50
Ontologi Menautkan API	53
InferID10cm	53
Kategori ICD-10-CM	53
Jenis ICD-10-CM	53
Ciri-ciri ICD-10-CM	54
Atribut ICD-10-CM	54
Kategori ekspresi waktu	54
Tipe	55
Jenis hubungan	55
Contoh masukan dan respons	55
InferRxNorm	58
Pemberitahuan penting	1
RxNorm kategori	58
RxNorm jenis	59
RxNorm atribut	59
RxNorm sifat	59
Contoh masukan dan respons	59

InfersNomeDCT	62
Kategori anatomi	63
Kategori kondisi medis	63
Kategori tes, perawatan, dan prosedur	64
Detail CT SNOMED	65
Contoh masukan dan respons	65
Ontologi yang menghubungkan analisis batch	73
Melakukan analisis batch	73
Kebijakan IAM	74
File keluaran analisis Batch	76
Keamanan	79
Perlindungan Data	80
Kontrol autentikasi dan akses	81
Autentikasi	81
Kontrol Akses	82
Gambaran umum pengelolaan akses	82
Menggunakan kebijakan berbasis identitas (kebijakan IAM) untuk Amazon Comprehend Medical	85
Referensi Amazon Comprehend Medical API	93
Logging panggilan API Amazon Comprehend Medical menggunakan AWS CloudTrail	93
Informasi Amazon Comprehend Medical di CloudTrail	94
Memahami entri file log Amazon Comprehend Medical	95
Validasi Kepatuhan	96
Ketangguhan	97
Keamanan Infrastruktur	97
Pedoman dan kuota	99
Pemberitahuan penting	99
Wilayah yang didukung	99
Throttling	99
Kuota keseluruhan	99
Riwayat dokumen	107
Amazon Comprehend Medical log perubahan	115
Perubahan pada operasi Amazon Comprehend Medical API	115
Perubahan pada operasi Amazon Comprehend Medical API	116
Perubahan pada operasi Amazon Comprehend Medical API	116
Perubahan pada operasi Amazon Comprehend Medical API	116

Perubahan pada operasi Amazon Comprehend Medical API	117
Perubahan pada operasi Amazon Comprehend Medical API	117
Perubahan pada operasi API Amazon Comprehend Medical InfersNomeDCT	117
Perubahan pada operasi Amazon Comprehend Medical API	117
.....	cxxiii

Apa itu Amazon Comprehend Medical?

Amazon Comprehend Medical mendeteksi dan mengembalikan informasi yang berguna dalam teks klinis yang tidak terstruktur seperti catatan dokter, ringkasan pelepasan, hasil tes, dan catatan kasus. Amazon Comprehend Medical menggunakan model natural language processing (NLP) untuk mendeteksi entitas, yang merupakan referensi tekstual untuk informasi medis seperti kondisi medis, obat-obatan, atau Informasi Kesehatan yang Dilindungi (PHI). Untuk daftar lengkap entitas yang terdeteksi, lihat [Mendeteksi entitas \(Versi 2\)](#). Amazon Comprehend Medical juga memungkinkan pengguna untuk menghubungkan entitas yang terdeteksi ini ke basis pengetahuan medis standar seperti RxNorm dan ICD-10-CM melalui operasi penautan ontologi.

Informasi dalam panduan pengembang ini ditujukan untuk pengembang aplikasi. Panduan ini mencakup informasi tentang penggunaan Amazon Comprehend Medical secara terprogram melalui AWS CLI atau Amazon Comprehend Medical API.

Harga untuk Amazon Comprehend Medical berbeda dari harga Amazon Comprehend. Untuk informasi selengkapnya, lihat [Amazon Comprehend Medical Pricing](#).

Bahasa yang Didukung

Amazon Comprehend Medical hanya mendeteksi entitas medis dalam teks bahasa Inggris (US-EN).

Pemberitahuan penting

Amazon Comprehend Medical bukan pengganti saran medis profesional, diagnosis, atau perawatan. Amazon Comprehend Medical memberikan skor kepercayaan yang menunjukkan tingkat kepercayaan pada keakuratan entitas yang terdeteksi. Identifikasi ambang kepercayaan yang tepat untuk kasus penggunaan Anda, dan gunakan ambang kepercayaan tinggi dalam situasi yang membutuhkan akurasi tinggi. Dalam kasus penggunaan tertentu, hasil harus ditinjau dan diverifikasi oleh pengulas manusia yang terlatih dengan tepat. Misalnya, Amazon Comprehend Medical hanya boleh digunakan dalam skenario perawatan pasien setelah ditinjau untuk akurasi dan penilaian medis yang baik oleh profesional medis terlatih.

Amazon Comprehend Medical kasus penggunaan

Anda dapat menggunakan Amazon Comprehend Medical untuk aplikasi perawatan kesehatan berikut:

- **Manajemen dan hasil kasus pasien** — Dokter dan penyedia layanan kesehatan dapat mengelola dan dengan mudah mengakses informasi medis yang tidak sesuai dengan bentuk tradisional. Pasien dapat melaporkan masalah kesehatan mereka dalam narasi dengan lebih banyak informasi daripada format standar. Dengan menganalisis catatan kasus, penyedia dapat mengidentifikasi kandidat untuk skrining dini kondisi medis sebelum kondisi menjadi lebih sulit dan mahal untuk diobati.
- **Penelitian klinis** — Ilmu kehidupan dan organisasi penelitian dapat mengoptimalkan proses pencocokan untuk mendaftarkan pasien ke dalam uji klinis. Dengan menggunakan Amazon Comprehend Medical untuk mendeteksi informasi terkait dalam teks klinis, peneliti dapat meningkatkan farmakovigilans, melakukan pengawasan pasca-pasar untuk memantau efek samping obat, dan menilai efektivitas terapeutik dengan mudah mendeteksi informasi penting dalam catatan tindak lanjut dan teks klinis lainnya. Misalnya, dapat lebih mudah dan lebih efektif untuk memantau bagaimana pasien merespons terapi tertentu dengan menganalisis narasi mereka.
- **Penagihan medis dan manajemen siklus pendapatan perawatan kesehatan** — Pembayar dapat memperluas analitik mereka untuk memasukkan dokumen yang tidak terstruktur seperti catatan klinis. Informasi lebih lanjut tentang diagnosis dapat dianalisis dan digunakan untuk membantu menentukan kode penagihan yang sesuai dari dokumen yang tidak terstruktur. Natural language processing (NLP) adalah komponen paling penting dari computer-assisted coding (CAC). Amazon Comprehend Medical menggunakan kemajuan terbaru dalam NLP untuk menganalisis teks klinis, membantu mengurangi waktu untuk pendapatan dan meningkatkan akurasi penggantian.
- **Ontologi menghubungkan** — Gunakan fitur penautan ontologi untuk mendeteksi entitas dari teks klinis dan menghubungkan entitas tersebut ke konsep standar dalam ontologi medis umum. InfericD10cm mengidentifikasi kemungkinan kondisi medis sebagai entitas. InfericD10CM menghubungkan entitas tersebut ke kode unik dari versi 2021 dari [Klasifikasi Penyakit Internasional, Revisi ke-10, Modifikasi Klinis \(ICD-10-CM\)](#). InferRxNorm mengidentifikasi obat yang tercantum dalam teks klinis sebagai entitas dan menghubungkan entitas tersebut ke pengidentifikasi konsep yang dinormalisasi dari [RxNormdatabase dari Perpustakaan Kedokteran Nasional AS](#). InferredNoMEDCT mendeteksi konsep medis seperti kondisi medis dan anatomi, tes medis, atau perawatan dan prosedur, sebagai entitas dan menghubungkannya dengan kode dari ontologi [Systematized Nomenclature of Medicine, Clinical Terms \(SNOMED CT\)](#).

Manfaat Amazon Comprehend Medical

Beberapa manfaat menggunakan Amazon Comprehend Medical meliputi:

- Integrasi pemrosesan bahasa alami yang mudah dan kuat ke dalam aplikasi Anda — Gunakan API untuk membangun kemampuan analisis teks ke dalam aplikasi Anda untuk pemrosesan bahasa alami yang kuat dan akurat.
- Akurasi — Gunakan teknologi pembelajaran mendalam untuk menganalisis teks secara akurat. Model kami terus dilatih dengan data baru di beberapa domain untuk meningkatkan akurasi.
- Skalabilitas — Mendeteksi informasi dari beberapa dokumen, memungkinkan wawasan cepat tentang kesehatan dan perawatan pasien.
- Integrasikan dengan layanan AWS lainnya — Amazon Comprehend Medical dirancang untuk bekerja secara mulus dengan layanan AWS lainnya seperti Amazon S3 dan AWS Lambda. Simpan dokumen Anda di Amazon S3, analisis data real-time dengan Kinesis Data Firehose, atau gunakan Amazon Transcribe untuk menyalin narasi pasien ke dalam teks yang dapat dianalisis oleh Amazon Comprehend Medical. Support for AWS Identity and Access Management (IAM) memudahkan untuk mengontrol akses ke operasi Amazon Comprehend Medical dengan aman. Menggunakan IAM, Anda dapat membuat dan mengelola pengguna dan grup AWS untuk memberikan akses yang sesuai untuk developer dan pengguna akhir Anda.
- Biaya rendah — Hanya membayar untuk dokumen yang Anda analisis. Tidak ada biaya minimum atau pun komitmen di muka

Kepatuhan HIPAA

Ini adalah Layanan yang Memenuhi Syarat HIPAA. [Untuk informasi lebih lanjut tentang AWS, Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan AS tahun 1996 \(HIPAA\), dan menggunakan AWS layanan untuk memproses, menyimpan, dan mengirimkan informasi kesehatan yang dilindungi \(PHI\), lihat Ikhtisar HIPAA.](#)

Koneksi ke Amazon Comprehend Medical yang mengandung PHI harus dienkripsi. Secara default, semua koneksi ke Amazon Comprehend Medical menggunakan HTTPS melalui TLS. Amazon Comprehend Medical tidak terus-menerus menyimpan konten pelanggan. Oleh karena itu, Anda tidak perlu mengkonfigurasi enkripsi saat istirahat dalam layanan.

Mengakses Amazon Comprehend Medical

1. AWS Management Console — Menyediakan antarmuka web yang dapat Anda gunakan untuk mengakses Amazon Comprehend Medical.
2. AWS Command Line Interface (AWS CLI) — Menyediakan perintah untuk serangkaian layanan AWS yang luas, termasuk Amazon Comprehend Medical, dan didukung di Windows, macOS, dan

Linux. Untuk informasi selengkapnya tentang menginstal AWS CLI, lihat [AWS Command Line Interface](#).

3. **AWS SDK** — AWS menyediakan SDK (perangkat pengembangan perangkat lunak) yang terdiri dari pustaka dan kode sampel untuk berbagai bahasa dan platform pemrograman (Java, Python, Ruby, .NET, iOS, Android, dll.). SDK menyediakan cara mudah untuk membuat akses terprogram ke Amazon Comprehend Medical dan AWS. Untuk informasi selengkapnya, lihat [AWS SDK](#).

Cara Memulai Amazon Comprehend Medical

Jika Anda adalah pengguna pertama kali Amazon Comprehend Medical, kami sarankan Anda membaca bagian berikut secara berurutan:

1. [Bagaimana Amazon Comprehend Medical bekerja](#)— Bagian ini memperkenalkan konsep Amazon Comprehend Medical.
2. [Memulai dengan Amazon Comprehend Medical](#)— Bagian ini menjelaskan cara mengatur akun Anda dan menguji Amazon Comprehend Medical.

Bagaimana Amazon Comprehend Medical bekerja

Amazon Comprehend Medical menggunakan model pemrosesan bahasa alami (NLP) yang telah terlatih untuk menganalisis teks klinis yang tidak terstruktur melalui deteksi entitas. Entitas adalah referensi tekstual untuk informasi medis seperti kondisi medis, obat-obatan, atau Informasi Health Terlindungi (PHI). Beberapa operasi melangkah lebih jauh dengan mendeteksi entitas dan kemudian menghubungkan entitas tersebut ke ontologi standar. Model ini terus dilatih pada sejumlah besar teks medis, sehingga Anda tidak perlu memberikan data pelatihan. Semua hasil mencakup skor kepercayaan, yang menunjukkan kepercayaan bahwa Amazon Comprehend Medical memiliki keakuratan entitas yang terdeteksi.

Deteksi entitas dan penautan ontologi dapat dilakukan baik sebagai operasi sinkron atau asinkron:

- Operasi sinkron- Memungkinkan analisis pada dokumen tunggal yang mengembalikan hasil analisis langsung ke aplikasi Anda. Gunakan operasi dokumen tunggal saat Anda membuat aplikasi interaktif yang bekerja pada satu dokumen sekaligus.
- Operasi asinkron — Mengaktifkan analisis pengumpulan atau kumpulan dokumen yang disimpan dalam bucket Amazon S3. Hasil analisis dikembalikan dalam bucket S3.

Note

Amazon Comprehend Medical hanya dapat menganalisis teks dalam bahasa Inggris (AS-EN).

Deteksi entitas sinkron

Parameter `DetectEntitiesV2` dan `DetectPhrases` mendeteksi entitas dalam teks klinis yang tidak terstruktur dari dokumen individual. Anda mengirim dokumen ke layanan Amazon Comprehend Medical dan menerima hasil analisis dalam tanggapan.

Analisis batch asinkron

Parameter `StartEntitiesDetectionV2Job` dan `StartPhrasesDetectionJob` memulai pekerjaan asinkron untuk mendeteksi referensi informasi medis seperti kondisi medis, perawatan, pengujian, dan

hasil atau informasi kesehatan yang dilindungi yang disimpan dalam bucket Amazon S3. Output dari pekerjaan pendeteksian ditulis ke bucket Amazon S3 terpisah yang dapat digunakan untuk pemrosesan lebih lanjut atau analisis hilir.

`ParameterStarticD10CminferenceJob`, dan `StarttrXNorminferenceJob` operasi mulai ontologi menghubungkan operasi batch yang mendeteksi entitas dan menghubungkan entitas tersebut ke kode standar dalam basis pengetahuan RxNorm dan ICD-10-CM.

Menghubungkan Ontologi

`ParameterInferiCD10cm`, `InfersNomedCT`, dan `InferRxNorm` operasi mendeteksi kondisi medis potensial dan obat-obatan dan menghubungkan mereka ke kode di ICD-10-CM, SNOMED CT, atau RxNorm basis pengetahuan, masing-masing. Anda dapat menggunakan ontologi menghubungkan analisis batch untuk menganalisis koleksi dokumen atau satu dokumen besar. Dengan menggunakan konsol atau ontologi menghubungkan API batch, Anda dapat melakukan operasi untuk memulai, menghentikan, daftar, dan menjelaskan pekerjaan analisis batch yang sedang berlangsung.

Menghubungkan ke konsep dalam basis pengetahuan ICD-10-CM tentang kondisi medis

`ParameterInferiCD10cm` operasi mendeteksi kondisi medis potensial dan menghubungkannya dengan kode dari versi 2019 Klasifikasi Penyakit Internasional, Revisi ke-10, Modifikasi Klinis (ICD-10-CM). Untuk setiap kondisi medis potensial yang terdeteksi, Amazon Comprehend Medical mencantumkan kode dan deskripsi ICD-10-CM yang cocok. `Liconditionshasil` termasuk skor kepercayaan diri, yang menunjukkan kepercayaan yang dimiliki Amazon Comprehend Medical ke akuratan entitas dengan konsep yang cocok dalam hasil.

Menghubungkan ke konsep di basis pengetahuan RxNorm obat

`ParameterInferRxNorm` operasi mengidentifikasi obat yang tercantum dalam catatan pasien sebagai entitas. Ini menghubungkan entitas ke pengidentifikasi konsep (`RxCui`) dari database RxNorm dari National Library of Medicine. Setiap `RxCui` unik untuk kekuatan dan bentuk dosis yang berbeda. `Listmedicationshasil` termasuk skor kepercayaan diri, yang menunjukkan kepercayaan yang dimiliki Amazon Comprehend Medical ke akuratan entitas yang cocok dengan konsep dari RxNorm basis pengetahuan. Amazon Comprehend Medical mencantumkan `RXCuis` teratas yang berpotensi cocok untuk setiap obat yang dideteksi dalam urutan menurun berdasarkan skor kepercayaan.

Menghubungkan ke konsep dalam basis pengetahuan SNOMED CT konsep medis

Operasi `InfersNomedCT` mengidentifikasi kemungkinan konsep medis sebagai entitas dan menghubungkannya ke kode dari versi 2021-03 dari Nomenklatur Kedokteran Sistematis, Istilah Klinis (SNOMED CT). SNOMED CT menyediakan kosakata komprehensif konsep medis, termasuk kondisi medis dan anatomi, serta tes medis, perawatan, dan prosedur. Untuk setiap ID konsep yang cocok, Amazon Comprehend Medical mengembalikan lima konsep medis teratas, masing-masing dengan skor kepercayaan diri dan informasi kontekstual seperti sifat dan atribut. ID konsep CT SNOMED kemudian dapat digunakan untuk menyusun data klinis pasien untuk pengkodean medis, pelaporan, atau analisis klinis bila digunakan dengan polyhierarki CT SNOMED.

Memulai dengan Amazon Comprehend Medical

Untuk mulai menggunakan Amazon Comprehend Medical, siapkan akun AWS dan buat pengguna AWS Identity and Access Management (IAM). Untuk menggunakan Amazon Comprehend Medical CLI, unduh dan konfigurasi.

Topik

- [Langkah 1: Menyiapkan akun AWS dan buat pengguna administrator](#)
- [Langkah 2: Menyiapkan AWS Command Line Interface \(AWS CLI\)](#)
- [Langkah 3: Memulai menggunakan konsol Amazon Comprehend Medical](#)
- [Langkah 4: Memulai menggunakan API Amazon Comprehend Medical](#)

Langkah 1: Menyiapkan akun AWS dan buat pengguna administrator

Sebelum Anda menggunakan Amazon Comprehend Medical untuk pertama kali, selesaikan tugas berikut:

1. [Daftar ke AWS](#)
2. [Membuat Pengguna IAM](#)

Daftar ke AWS

Saat Anda mendaftar ke Amazon Web Services (AWS), akun AWS Anda secara otomatis terdaftar untuk semua layanan AWS, termasuk Amazon Comprehend Medical. Anda hanya dikenakan biaya untuk layanan yang Anda gunakan.

Dengan Amazon Comprehend Medical, Anda hanya membayar untuk sumber daya yang Anda gunakan. Jika Anda adalah pelanggan baru AWS, Anda dapat memulai dengan Amazon Comprehend Medical secara gratis. Untuk informasi selengkapnya, lihat [Tingkatan Gratis AWS](#).

Jika Anda sudah memiliki akun AWS, lewati ke bagian berikutnya.

Untuk membuat akun AWS

1. Buka <https://portal.aws.amazon.com/billing/signup>.

2. Ikuti petunjuk online.

Bagian dari prosedur pendaftaran melibatkan menerima panggilan telepon dan memasukkan kode verifikasi di keypad telepon.

Ketika Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWS dibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik terbaik keamanan, [tetapkan akses administratif ke pengguna administratif](#), dan gunakan hanya pengguna root untuk melakukan [tugas yang memerlukan akses pengguna root](#).

Catat ID akun AWS Anda karena Anda akan membutuhkannya untuk tugas berikutnya.

Membuat Pengguna IAM

Layanan di AWS, seperti Amazon Comprehend Medical, mewajibkan Anda memberikan kredensial saat Anda mengaksesnya. Layanan kemudian dapat menentukan apakah Anda memiliki izin untuk mengakses sumber daya layanan.

Kami sangat menyarankan Anda untuk mengakses AWS menggunakan AWS Identity and Access Management (IAM), bukan kredensial untuk akun AWS Anda. Untuk menggunakan IAM untuk mengakses AWS, buat pengguna IAM, tambahkan pengguna ke grup IAM dengan izin administratif, lalu berikan izin administratif ke pengguna IAM. Anda kemudian dapat mengakses AWS menggunakan URL khusus dan kredensial pengguna IAM.

Latihan Memulai dalam panduan ini mengasumsikan Anda memiliki pengguna dengan hak istimewa administrator `adminuser`.

Untuk membuat administrator dan masuk ke konsol tersebut

1. Buat pengguna bernama `adminuser` di akun AWS Anda. Untuk melihat instruksi, buka [Membuat Grup Pengguna dan Administrator IAM Pertama Anda](#) di Panduan Pengguna IAM.
2. Masuk ke AWS Management Console menggunakan URL khusus. Untuk informasi selengkapnya, lihat [Cara Pengguna Masuk ke Akun Anda](#) dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang IAM, lihat hal berikut:

- [AWS Identity and Access Management \(IAM\)](#)
- [Memulai](#)

- [Panduan Pengguna IAM](#)

Langkah selanjutnya

[Langkah 2: Menyiapkan AWS Command Line Interface \(AWS CLI\)](#)

Langkah 2: Menyiapkan AWS Command Line Interface (AWS CLI)

Anda tidak memerlukan AWS CLI untuk melakukan langkah-langkah dalam latihan Memulai. Namun, beberapa latihan lain dalam panduan ini memang membutuhkannya. Jika Anda mau, Anda dapat melewati langkah ini dan langsung ke [Langkah 3: Memulai menggunakan konsol Amazon Comprehend Medical](#), lalu menyiapkan AWS CLI nanti.

Untuk mengatur AWS CLI

1. Unduh dan konfigurasi AWS CLI. Untuk melakukannya, lihat topik berikut di Panduan Pengguna AWS Command Line Interface:
 - [Mendapatkan Set Up dengan AWS Command Line Interface](#)
 - [Mengkonfigurasi AWS Command Line Interface](#)
2. Dalam file AWS CLI config, tambahkan profil bernama untuk administrator:

```
[profile adminuser]
aws_access_key_id = adminuser access key ID
aws_secret_access_key = adminuser secret access key
region = aws-region
```

Anda menggunakan profil ini saat menjalankan perintah AWS CLI. Untuk informasi selengkapnya tentang profil yang diberi nama, lihat [Profil yang Diberi Nama](#) dalam Panduan Pengguna AWS Command Line Interface. Untuk daftar AWS Wilayah, lihat [Wilayah dan Titik Akhir](#) di bagian Referensi Umum Amazon Web Services.

3. Verifikasikan penyiapan dengan mengetikkan perintah bantuan berikut pada command prompt:

```
aws help
```


Langkah selanjutnya

[Langkah 3: Memulai menggunakan konsol Amazon Comprehend Medical](#)

Langkah 3: Memulai menggunakan konsol Amazon Comprehend Medical

Cara termudah untuk mulai menggunakan konsol Comprehend Medical adalah dengan menganalisis file teks pendek. Jika Anda belum meninjau konsep dan terminologi di [Bagaimana Amazon Comprehend Medical bekerja](#), kami menyarankan Anda untuk melakukan hal itu sebelum melanjutkan.

Topik

- [Menganalisis teks klinis menggunakan konsol](#)

Menganalisis teks klinis menggunakan konsol

Konsol Medical Comprehend memungkinkan Anda menganalisis isi teks klinis, hingga 20.000 karakter panjang. Hasilnya ditunjukkan di konsol sehingga Anda dapat meninjau analisisnya.

Untuk mulai menganalisis dokumen, masuk keAWS Management Console dan buka konsol Medical Comprehend.

Di bawah Comprehend Medical, pilih Analisis waktu nyata.

Konsol menampilkan teks sampel dan analisis teks itu:

Real-time analysis [Info](#)

See how Comprehend Medical recognizes entities related to the healthcare domain. To analyze your text, type or paste it in the text box.

Input text

[Supported languages](#)

Pt is 87 yo woman, highschool teacher with past medical history that includes
 - status post cardiac catheterization in April 2019.
 She presents today with palpitations and chest pressure.
 HPI : Sleeping trouble on present dosage of Clonidine. Severe Rash on face and leg, slightly itchy
 Meds : Vyvanse 50 mgs po at breakfast daily,
 Clonidine 0.2 mgs -- 1 and 1 / 2 tabs po qhs
 HEENT : Boggy inferior turbinates, No oropharyngeal lesion

415 of 20000 characters used.

Clear text Analyze

Anda dapat mengganti teks sampel dengan teks Anda sendiri dalam bahasa Inggris dan kemudian memilih menganalisis untuk mendapatkan analisis teks Anda.

Insights [Info](#)

Entities | RxNorm concepts | ICD-10-CM concepts | SNOMED CT concepts

Analyzed text

Pt is **87** yo woman, **highschool teacher** with past medical history that includes

- Age (87)
- Dx name (highschool teacher)
- Profession (highschool teacher)

- **status post** **cardiac catheterization** in **April 2019**.

- Time to procedure name (status post)
- Procedure name (cardiac catheterization)
- Time to procedure name (April 2019)
- Date (April 2019)

She presents today with **palpitations** and **chest pressure**.

- Dx name (palpitations)
- System organ site (chest)

Relationships: Overlap between status post and cardiac catheterization; Overlap between cardiac catheterization and April 2019; Sign between palpitations and chest pressure.

Di bawah teks input, teks yang dianalisis adalah kode warna untuk menunjukkan kategori entitas:

- Tag oranye mengidentifikasi data PHI.


- Tag merah mengidentifikasi Obat.
- Tag hijau mengidentifikasi Kondisi Medis.
- Tag biru mengidentifikasi Uji, Pengobatan, atau Prosedur (TTP).
- Tag ungu mengidentifikasi Anatomi.
- Tag merah muda mengidentifikasi Ekspresi Waktu.


Untuk informasi selengkapnya, lihat [Bagaimana Amazon Comprehend Medical bekerja](#).

Di konsol, di bawah kotak input, Tekspane menunjukkan informasi lebih lanjut tentang teks.

ParameterEntitasbagian menampilkan kartu untuk entitas yang ditemukan dalam teks:

▼ Results (27)

Find entities All ▼ < 1 2 3 4 > 

Entity	Type	Category	Traits
87 0.9997 score	● Age	Protected health information	-
highschool teacher 0.7305 score	● Dx name	Medical condition	-
highschool teacher 0.2063 score	● Profession	Protected health information	-
 status post 0.9525 score	● Time to procedure name	Time expression	-
cardiac catheterization 0.8982 score	● Procedure name	Test treatment procedure	-

Setiap kartu menunjukkan teks dan jenis entitas.

Di samping masing-masing entitas, skor mewakili kepercayaan yang dimiliki Comprehend Medical dalam identifikasi teks sebagai jenis entitas yang ditunjukkan.

Untuk melihat struktur JSON dari permintaan dan hasilnya, pilih Integrasi aplikasi. Struktur JSON sama dengan struktur yang dikembalikan oleh operasi.

Langkah Selanjutnya

[Langkah 4: Memulai menggunakan API Amazon Comprehend Medical](#)

Langkah 4: Memulai menggunakan API Amazon Comprehend Medical

Contoh-contoh berikut ini mendemonstrasikan cara menggunakan operasi Amazon Comprehend Medical menggunakan operasi AWS CLI, Java, dan Python. Gunakan mereka untuk mempelajari tentang operasi Amazon Comprehend Medical dan sebagai blok bangunan untuk aplikasi Anda sendiri.

Untuk menjalankan AWS CLI dan Python contoh, instal AWS CLI. Untuk informasi selengkapnya, lihat [Langkah 2: Menyiapkan AWS Command Line Interface \(AWS CLI\)](#).

Untuk menjalankan contoh Java, instal AWS SDK for Java. Untuk petunjuk pemasangan AWS SDK for Java, lihat [Mengatur AWS SDK for Java](#).

Topik

- [Mendeteksi entitas medis menggunakan AWS Command Line Interface](#)
- [Mendeteksi entitas medis menggunakan AWS SDK for Java](#)
- [Mendeteksi entitas medis menggunakan AWS SDK for Python \(Boto\)](#)

Mendeteksi entitas medis menggunakan AWS Command Line Interface

Contoh berikut menunjukkan menggunakan `DetectEntitiesV2` operasi menggunakan AWS CLI untuk mengembalikan entitas medis yang terdeteksi dalam teks. Untuk menjalankan contoh, Anda harus instal AWS CLI. Untuk informasi selengkapnya, lihat [the section called "Langkah 2: Siapkan AWS CLI"](#).

Contohnya diformat untuk Unix, Linux, dan macOS. Untuk Windows, ganti karakter kelanjutan Unix (`\`) pada akhir setiap baris dengan caret (`^`).

```
aws comprehendmedical detect-entities-v2 \  
  --endpoint endpoint \  
  --region region \  
  --text "aspirin is required 20 mg po daily for 2 times a day"
```

Respons adalah sebagai berikut ini:

```
{
  "Entities": [
    {
      "Category": "MEDICATION",
      "BeginOffset": 0,
      "EndOffset": 7,
      "Text": "aspirin",
      "Traits": [],
      "Score": 0.9988090991973877,
      "Attributes": [
        {
          "BeginOffset": 20,
          "EndOffset": 25,
          "Text": "20 mg",
          "Traits": [],
          "Score": 0.9559056162834167,
          "Type": "DOSAGE",
          "Id": 1,
          "RelationshipScore": 0.9981593489646912
        },
        {
          "BeginOffset": 26,
          "EndOffset": 28,
          "Text": "po",
          "Traits": [],
          "Score": 0.9995359182357788,
          "Type": "ROUTE_OR_MODE",
          "Id": 2,
          "RelationshipScore": 0.9969323873519897
        },
        {
          "BeginOffset": 29,
          "EndOffset": 34,
          "Text": "daily",
          "Traits": [],
          "Score": 0.9803128838539124,
          "Type": "FREQUENCY",
          "Id": 3,
          "RelationshipScore": 0.9990783929824829
        },
        {
          "BeginOffset": 39,
          "EndOffset": 46,
```

```

        "Text": "2 times",
        "Traits": [],
        "Score": 0.8623972535133362,
        "Type": "DURATION",
        "Id": 4,
        "RelationshipScore": 0.9996501207351685
    },
    {
        "BeginOffset": 50,
        "EndOffset": 53,
        "Text": "tab",
        "Traits": [],
        "Score": 0.784785270690918,
        "Type": "FORM",
        "Id": 5,
        "RelationshipScore": 0.9986748695373535
    }
],
    "Type": "GENERIC_NAME",
    "Id": 0
}
],
    "UnmappedAttributes": []
}

```

Mendeteksi entitas medis menggunakan AWS SDK for Java

Contoh berikut menggunakan `DetectEntitiesV2` operasi dengan Java. Untuk menjalankan contoh, instal AWS SDK for Java. Untuk petunjuk tentang menginstal AWS SDK for Java, lihat [Mengatur AWS SDK for Java](#).

```

import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.AWSCredentialsProvider;
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicAWSCredentials;
import com.amazonaws.client.builder.AwsClientBuilder;
import com.amazonaws.services.comprehendmedical.AWSComprehendMedical;
import com.amazonaws.services.comprehendmedical.AWSComprehendMedicalClient;
import com.amazonaws.services.comprehendmedical.model.DetectEntitiesRequest;
import com.amazonaws.services.comprehendmedical.model.DetectEntitiesResult;

public class SampleAPICall {

```

```

public static void main() {

    AWSCredentialsProvider credentials
        = new AWSStaticCredentialsProvider(new BasicAWSCredentials("YOUR AWS
ACCESS KEY", "YOUR AWS SECRET"));

    AWSComprehendMedical client = AWSComprehendMedicalClient.builder()

.withCredentials(credentials)

.withRegion("YOUR
REGION")

.build();

    DetectEntitiesV2Request request = new DetectEntitiesV2Request();
    request.setText("cerealx 84 mg daily");

    DetectEntitiesV2Result result = client.detectEntitiesV2(request);
    result.getEntities().forEach(System.out::println);
}
}

```

Output berisi tiga entitas yang ditemukan dalam teks input, lokasi mereka dalam teks input. Tingkat kepercayaan bahwa entitas diidentifikasi dengan benar juga terdaftar pada setiap entitas. Output menunjukkan `Generic_Name`, `Dosage`, dan `Frequency` entitas dari contoh sebelumnya.

```

{Id: 0, BeginOffset: 0, EndOffset: 3, Score: 0.9940211, Text: Bob, Category:
PROTECTED_HEALTH_INFORMATION, Type: NAME, Traits: [],}
{Id: 2, BeginOffset: 23, EndOffset: 30, Score: 0.99914634, Text: aspirin, Category:
MEDICATION, Type: GENERIC_NAME, Traits: [], Attributes:
[{Type: DOSAGE, Score: 0.9630807, RelationshipScore: 0.99969745, Id: 1, BeginOffset:
14, EndOffset: 19, Text: 50 mg, Traits: []}]}

```

Mendeteksi entitas medis menggunakan AWS SDK for Python (Boto)

Contoh berikut menggunakan `DetectEntitiesV2` operasi dengan Python. Untuk menjalankan sampel, instal AWS CLI. Untuk informasi selengkapnya, lihat [the section called “Langkah 2: Siapkan AWS CLI”](#).

```

import boto3
client = boto3.client(service_name='comprehendmedical', region_name='YOUR REGION')

```

```
result = client.detect_entities(Text= 'cerealx 84 mg daily')
entities = result['Entities'];
for entity in entities:
    print('Entity', entity)
```

Output berisi tiga entitas yang ditemukan dalam teks input, lokasi mereka dalam teks input. Tingkat kepercayaan bahwa entitas diidentifikasi dengan benar juga terdaftar pada setiap entitas. Output menunjukkan `Generic_Name`, `Dosage`, dan `Frequency` entitas dari contoh sebelumnya.

```
('Entity', {'Category': 'MEDICATION', 'BeginOffset': 0, 'EndOffset': 7,
            'Text': 'cerealx', 'Traits': [], 'Score': 0.8877691626548767,
            'Attributes': [{'BeginOffset': 8, 'EndOffset': 13,
                            'Text': '84 mg', 'Traits': [], 'Score': 0.9337134957313538, 'Type':
            'DOSAGE', 'Id': 1, 'RelationshipScore': 0.9995118379592896},
                            {'BeginOffset': 14, 'EndOffset': 19, 'Text': 'daily', 'Traits': [],
            'Score': 0.990627646446228, 'Type': 'FREQUENCY',
            'Id': 2, 'RelationshipScore': 0.9987651109695435}], 'Type':
            'BRAND_NAME', 'Id': 0})
```


Amazon Comprehend Medical dan VPC endpoint antarmuka (AWS PrivateLink)

Anda dapat membangun koneksi privat antara VPC Anda dan Amazon Comprehend Medical dengan membuat VPC endpoint antarmuka. Titik akhir VPC antarmuka didukung oleh [AWS PrivateLink](#), teknologi yang dapat Anda gunakan untuk mengakses API Amazon Comprehend Medical tanpa gateway internet, perangkat NAT, koneksi VPN, atau AWS Direct Connect koneksi. Instans dalam VPC Anda tidak memerlukan alamat IP publik untuk berkomunikasi dengan API Amazon Comprehend Medical. Lalu lintas antara VPC Anda dan Amazon Comprehend Medical tidak meninggalkan jaringan Amazon.

Setiap titik akhir antarmuka diwakili oleh satu atau lebih [Antarmuka Jaringan Elastis](#) dalam subnet Anda.

Untuk informasi lebih lanjut, lihat [VPC endpoint Antarmuka \(AWS PrivateLink\)](#) dalam Panduan Pengguna Amazon VPC.

Pertimbangan untuk VPC endpoint Amazon Comprehend Medical

Sebelum Anda menyiapkan VPC endpoint antarmuka untuk Amazon Comprehend Medical, pastikan bahwa Anda meninjau [Properti endpoint antarmuka dan keterbatasan](#) di Panduan Pengguna Amazon VPC.

Amazon Comprehend Medical mendukung panggilan ke semua tindakan API-nya dari VPC Anda.

Membuat VPC endpoint antarmuka untuk Amazon Comprehend Medical

Anda dapat membuat VPC endpoint untuk Amazon Comprehend Medical menggunakan konsol Amazon VPC atau AWS Command Line Interface (AWS CLI). Untuk informasi lebih lanjut, lihat [Membuat titik akhir antarmuka](#) di Panduan Pengguna Amazon VPC.

Buat VPC endpoint untuk Amazon Comprehend Medical menggunakan nama layanan berikut:

- `com.amazonaws.daerah.comprehendmedical`

Jika Anda mengaktifkan DNS pribadi untuk titik akhir, Anda dapat mengajukan permintaan API ke Amazon Comprehend Medical menggunakan nama DNS default untuk Wilayah. Misalnya, `comprehendmedical.us-east-1.amazonaws.com`.

Untuk informasi lebih lanjut, lihat [Mengakses layanan melalui titik akhir antarmuka](#) di Panduan Pengguna Amazon VPC.

Membuat kebijakan VPC endpoint untuk Amazon Comprehend Medical

Anda dapat melampirkan kebijakan titik akhir ke VPC endpoint yang mengontrol akses ke Amazon Comprehend Medical. Kebijakan menentukan informasi berikut ini:

- Prinsip-prinsip yang dapat melakukan tindakan.
- Tindakan yang dapat dilakukan.
- Sumber daya yang dapat digunakan untuk mengambil tindakan.

Untuk informasi selengkapnya, lihat [Mengendalikan akses ke layanan dengan titik akhir VPC](#) dalam Panduan Pengguna Amazon VPC.

Contoh: Kebijakan VPC endpoint untuk tindakan Amazon Comprehend Medical

Berikut ini adalah contoh kebijakan titik akhir untuk Amazon Comprehend Medical. Jika dilampirkan ke titik akhir, kebijakan ini memberikan akses ke Amazon Comprehend MedicalDetectEntitiesV2 tindakan untuk semua kepala sekolah pada semua sumber daya.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "comprehendmedical:DetectEntitiesV2"
      ],
      "Resource": "*"
    }
  ]
}
```

Operasi

Gunakan Amazon Comprehend Medical untuk memeriksa dokumen klinis dan untuk mendapatkan berbagai wawasan tentang konten mereka menggunakan model pemrosesan bahasa alami (NLP) pra-terlatih. Anda dapat melakukan analisis baik pada file tunggal atau sebagai analisis secara tidak dapat dilakukan di bucket Amazon Simple Storage Service (S3).

Dengan Amazon Comprehend Medical, Anda dapat melakukan hal berikut di dokumen Anda:

- [Mendeteksi entitas \(Versi 2\)](#)— Periksa teks klinis tidak terstruktur untuk mendeteksi referensi tekstual untuk informasi medis seperti kondisi medis, pengobatan, tes dan hasil, dan obat-obatan. Versi ini menggunakan model yang berbeda dari API entitas Detect asli, dan ada beberapa perubahan pada output.
- [Mendeteksi PHI](#) - Periksa teks klinis yang tidak terstruktur untuk mendeteksi referensi tekstual ke informasi kesehatan yang dilindungi (PHI) seperti nama dan alamat.
- [Mendeteksi entitas](#)— Periksa teks klinis tidak terstruktur untuk mendeteksi referensi tekstual untuk informasi medis seperti kondisi medis, pengobatan, tes dan hasil, dan obat-obatan. Gunakan `DetectEntitiesV2` untuk semua aplikasi baru.

Amazon Comprehend Medical mendukung beberapa operasi API yang dapat Anda gunakan untuk melakukan analisis teks batch pada dokumen klinis. Untuk mempelajari selengkapnya tentang cara menggunakan operasi ini, lihat [the section called “API batch analisis teks”](#).

Topik

- [Mendeteksi entitas \(Versi 2\)](#)
- [Mendeteksi entitas](#)
- [Mendeteksi PHI](#)
- [API batch analisis teks](#)

Mendeteksi entitas (Versi 2)

Gunakan `DetectEntitiesV2` untuk mendeteksi entitas dalam satu file atau `StartEntitiesDetectionV2Job` untuk analisis batch pada beberapa file. Anda dapat mendeteksi entitas dalam kategori berikut:

- **ANATOMY** : Mendeteksi referensi ke bagian-bagian tubuh atau sistem tubuh dan lokasi bagian-bagian atau sistem tersebut.
- **BEHAVIORAL_ENVIRONMENTAL_SOCIAL**: Mendeteksi perilaku dan kondisi di lingkungan yang berdampak pada kesehatan seseorang. Ini termasuk penggunaan tembakau, konsumsi alkohol, penggunaan narkoba rekreasi, alergi, jenis kelamin, dan ras/etnis.
- **MEDICAL_CONDITION**: Mendeteksi tanda-tanda, gejala, dan diagnosis kondisi medis.
- **MEDICATION**: Mendeteksi informasi obat dan dosis pada pasien.
- **PROTECTED_HEALTH_INFORMATION**: Mendeteksi informasi pribadi pasien.
- **TEST_TREATMENT_PROCEDURE** : Mendeteksi prosedur yang digunakan untuk menentukan kondisi medis.
- **TIME_EXPRESSION**: Mendeteksi entitas yang terkait dengan waktu ketika mereka dikaitkan dengan entitas terdeteksi.

Keenam kategori terdeteksi oleh operasi DetectEntitiesV2. Untuk analisis khusus untuk mendeteksi PHI, gunakan DetectPHI pada file tunggal dan startPHI untuk analisis batch. DetectionJob

Amazon Comprehend Medical mendeteksi informasi di kelas-kelas berikut:

- **Entitas**: Referensi teks untuk nama objek yang relevan, seperti orang, perawatan, obat-obatan, dan kondisi medis. Sebagai contoh, *ibuprofen*.
- **Kategori**: Pengelompokan umum yang dimiliki entitas. Misalnya, *ibuprofen* adalah bagian dari **MEDICATION** kategori.
- **Jenis**: Jenis entitas yang terdeteksi dalam satu kategori. Misalnya, *ibuprofen* ada dalam **GENERIC_NAME** tipe **MEDICATION** kategori.
- **Atribut**: Informasi yang terkait dengan suatu entitas, seperti dosis obat. Misalnya, *200 mg* adalah atribut dari entitas *ibuprofen*.
- **Sifat**: Sesuatu yang dipahami Amazon Comprehend Medical tentang suatu entitas, berdasarkan konteks. Misalnya, obat memiliki **NEGATION** sifat jika pasien tidak meminumnya.
- **Relationship Type**: Hubungan antara entitas dan atribut.

Amazon Comprehend Medical memberi Anda lokasi entitas dalam teks masukan. Di konsol Amazon Comprehend, ini menunjukkan lokasi secara grafis. Bila Anda menggunakan API, itu menunjukkan lokasi dengan offset numerik.

Setiap entitas dan atribut menyertakan skor yang menunjukkan tingkat kepercayaan yang dimiliki Amazon Comprehend Medical dalam keakuratan deteksi. Setiap atribut juga memiliki skor hubungan. Skor menunjukkan tingkat kepercayaan yang dimiliki Amazon Comprehend Medical dalam keakuratan hubungan antara atribut dan entitas induknya. Identifikasi ambang kepercayaan yang sesuai untuk kasus penggunaan Anda. Gunakan ambang batas kepercayaan tinggi dalam situasi yang membutuhkan akurasi tinggi. Filter data yang tidak memenuhi ambang batas.

Kategori anatomi

ANATOMYKategori mendeteksi referensi ke bagian-bagian tubuh atau sistem tubuh dan lokasi bagian-bagian atau sistem tersebut.

Tipe

- **SYSTEM_ORGAN_SITE**: Sistem tubuh, lokasi anatomi atau daerah, dan situs tubuh.

Atribut

- **DIRECTION**: Istilah Directional. Misalnya, kiri, kanan, medial, lateral, atas, bawah, posterior, anterior, distal, proksimal, kontralateral, bilateral, ipsilateral, dorsal, ventral, dan sebagainya.

Kategori perilaku, lingkungan, dan kesehatan sosial

BEHAVIORAL_ENVIRONMENTAL_SOCIALKategori ini mendeteksi referensi perilaku dan kondisi di lingkungan yang memengaruhi kesehatan seseorang.

Tipe

- **ALCOHOL_CONSUMPTION**: Mendefinisikan konsumsi alkohol pasien dalam hal status penggunaan, frekuensi, jumlah, dan durasi.
- **ALLERGIES**: Mendefinisikan alergi pasien dan tanggapan terhadap alergen.
- **GENDER**: Identifikasi karakteristik identitas gender.
- **RACE_ETHNICITY**: Konstruksi sosial-politik identifikasi pasien dengan kelompok ras dan etnis tertentu.
- **REC_DRUG_USE**: Mendefinisikan penggunaan obat rekreasi pasien dalam hal status penggunaan, frekuensi, jumlah, dan durasi.

- **TOBACCO_USE**: Mendefinisikan penggunaan tembakau pasien dalam hal status penggunaan, frekuensi, jumlah, dan durasi.

Atribut

Atribut terdeteksi berikut hanya berlaku untuk jenis **ALCOHOL_CONSUMPTION**, **TOBACCO_USE**, dan **REC_DRUG_USE**:

- **AMOUNT**: Jumlah alkohol, tembakau, atau obat rekreasi yang digunakan.
- **DURATION**: Berapa lama alkohol, tembakau, atau obat rekreasi telah digunakan.
- **FREQUENCY**: Seberapa sering alkohol, tembakau, atau obat rekreasi digunakan.

Sifat-sifat

Ciri-ciri terdeteksi berikut hanya berlaku untuk jenis **ALCOHOL_CONSUMPTION**, **ALLERGIESTOBACCO_USE**, dan **REC_DRUG_USE**:

- **NEGATION**: Indikasi bahwa hasil atau tindakan negatif atau tidak dilakukan.
- **PAST_HISTORY**: Indikasi bahwa penggunaan alkohol, tembakau, atau obat-obatan rekreasi berasal dari masa lalu pasien (sebelum pertemuan saat ini).

Kategori kondisi medis

MEDICAL_CONDITION Kategori ini mendeteksi tanda, gejala, dan diagnosis kondisi medis. Kategori ini memiliki satu jenis entitas, empat atribut, dan empat sifat. Satu atau lebih sifat dapat dikaitkan dengan suatu tipe. Informasi kontekstual tentang atribut dan hubungannya dengan diagnosis terdeteksi dan dipetakan ke **DX_NAME** melalui **RELATIONSHIP_EXTRACTION**. Misalnya, dari teks “nyeri kronis di kaki kiri”, “kronis” terdeteksi sebagai atribut **ACUITY**, “kiri” terdeteksi sebagai atribut **DIRECTION**, dan “kaki” terdeteksi sebagai atribut **SYSTEM_ORGAN_SITE**. Hubungan masing-masing atribut ini dipetakan ke entitas kondisi medis “rasa sakit,” bersama dengan skor kepercayaan diri.

Tipe

- **DX_NAME**: Semua kondisi medis terdaftar. **DX_NAME** Tipe ini termasuk penyakit saat ini, alasan kunjungan, dan riwayat kesehatan.

Atribut

- **ACUITY**: Penentuan contoh penyakit, seperti kronis, akut, mendadak, persisten, atau bertahap.
- **DIRECTION**: Istilah Directional. Misalnya, kiri, kanan, medial, lateral, atas, bawah, posterior, anterior, distal, proksimal, kontralateral, bilateral, ipsilateral, dorsal, atau ventral.
- **SYSTEM_ORGAN_SITE**: Lokasi anatomi.
- **QUALITY**: Setiap istilah deskriptif dari kondisi medis, seperti tahap atau kelas.

Sifat-sifat

- **DIAGNOSIS**: Kondisi medis yang ditentukan sebagai penyebab atau akibat gejala. Gejala dapat ditemukan melalui temuan fisik, laporan laboratorium atau radiologis, atau cara lain.
- **HYPOTHETICAL**: Indikasi bahwa kondisi medis dinyatakan sebagai hipotesis.
- **LOW_CONFIDENCE**: Indikasi bahwa kondisi medis dinyatakan memiliki ketidakpastian yang tinggi. Ini tidak terkait langsung dengan skor kepercayaan yang diberikan.
- **NEGATION**: Indikasi bahwa hasil atau tindakan negatif atau tidak dilakukan.
- **PERTAINS_TO_FAMILY**: Indikasi bahwa kondisi medis relevan dengan keluarga pasien, bukan pasien.
- **SIGN**: Kondisi medis yang dilaporkan dokter.
- **SYMPTOM**: Kondisi medis yang dilaporkan pasien.

Kategori obat

MEDICATION Kategori mendeteksi informasi pengobatan dan dosis untuk pasien. Satu atau lebih atribut dapat diterapkan ke tipe.

Tipe

- **BRAND_NAME**: Nama merek obat atau agen terapeutik yang dilindungi hak cipta.
- **GENERIC_NAME**: Nama non-merek, nama bahan, atau campuran formula obat atau agen terapeutik.

Atribut

- **DOSAGE**: Jumlah obat yang dipesan.

- **DURATION:** Berapa lama obat harus diberikan.
- **FORM:** Bentuk obatnya.
- **FREQUENCY:** Seberapa sering mengelola obat.
- **RATE:** Tingkat pemberian obat (terutama untuk infus obat atau infus).
- **ROUTE_OR_MODE:** Metode pemberian obat.
- **STRENGTH:** Kekuatan obat.

Sifat-sifat

- **NEGATION:** Setiap indikasi bahwa pasien tidak minum obat.
- **PAST_HISTORY:** Indikasi bahwa obat yang terdeteksi berasal dari masa lalu pasien (sebelum pertemuan saat ini).

Kategori informasi kesehatan yang dilindungi

PROTECTED_HEALTH_INFORMATION Kategori mendeteksi informasi pribadi pasien. Lihat [Mendeteksi PHI](#) untuk mempelajari lebih lanjut tentang operasi ini.

Tipe

- **ADDRESS:** Semua subdivisi geografis dari alamat fasilitas, unit, atau bangsal apa pun dalam suatu fasilitas.
- **AGE:** Semua komponen usia, rentang usia, atau usia yang disebutkan. Ini termasuk pasien, anggota keluarga, atau orang lain. Defaultnya adalah dalam beberapa tahun, kecuali dinyatakan lain.
- **EMAIL:** Alamat email apa pun.
- **ID:** Nomor Jaminan Sosial, nomor rekam medis, nomor identifikasi fasilitas, nomor uji klinis, nomor sertifikat atau lisensi, nomor kendaraan atau perangkat, tempat perawatan, atau penyedia. Ini juga mencakup jumlah biometrik pasien, seperti tinggi badan, berat badan, atau nilai laboratorium.
- **NAME:** Semua nama. Biasanya, nama pasien, keluarga, atau penyedia.
- **PHONE_OR_FAX:** Telepon, faks, atau nomor pager apa pun. Tidak termasuk nomor telepon bernama, seperti 1-800-QUIT-NOW dan 911.
- **PROFESSION:** Setiap profesi atau majikan yang berkaitan dengan pasien atau keluarga pasien. Itu tidak termasuk profesi dokter yang disebutkan dalam catatan.

Kategori tes, perawatan, dan prosedur

TEST_TREATMENT_PROCEDURE Kategori mendeteksi prosedur yang digunakan untuk menentukan kondisi medis. Satu atau lebih atribut dapat dikaitkan dengan entitas dari **TEST_NAME** jenis.

Tipe

- **PROCEDURE_NAME**: Intervensi sebagai tindakan satu kali yang dilakukan pada pasien untuk mengobati kondisi medis atau untuk memberikan perawatan pasien.
- **TEST_NAME**: Prosedur yang dilakukan pada pasien untuk diagnostik, pengukuran, skrining, atau penilaian yang mungkin memiliki nilai yang dihasilkan. Ini termasuk prosedur, proses, evaluasi, atau penilaian apa pun untuk menentukan diagnosis, untuk mengesampingkan atau menemukan suatu kondisi, atau untuk menskalakan atau menilai pasien.
- **TREATMENT_NAME**: Intervensi dilakukan selama rentang waktu untuk memerangi penyakit atau gangguan. Ini termasuk pengelompokan obat, seperti antivirus dan vaksinasi.

Atribut

- **TEST_VALUE**: Hasil tes. Berlaku hanya untuk jenis **TEST_NAME** entitas.
- **TEST_UNIT**: Satuan ukuran yang mungkin menyertai nilai tes. Berlaku hanya untuk jenis **TEST_NAME** entitas.

Sifat-sifat

- **FUTURE**: Indikasi bahwa tes, perawatan, atau prosedur mengacu pada tindakan atau peristiwa yang akan terjadi setelah subjek catatan.
- **HYPOTHETICAL**: Indikasi bahwa tes, pengobatan, atau prosedur dinyatakan sebagai hipotesis.
- **NEGATION**: Indikasi bahwa hasil atau tindakan negatif atau tidak dilakukan.
- **PAST_HISTORY**: Indikasi bahwa tes, pengobatan, atau prosedur berasal dari masa lalu pasien (sebelum pertemuan saat ini).

Kategori ekspresi waktu

TIME_EXPRESSION Kategori mendeteksi entitas yang terkait dengan waktu. Ini termasuk entitas seperti tanggal dan ekspresi waktu seperti “tiga hari yang lalu,” “hari ini,” “saat ini,” “hari masuk,”

“bulan lalu,” atau “16 hari.” Hasil dalam kategori ini hanya dikembalikan jika dikaitkan dengan entitas. Misalnya, “Kemarin, pasien mengambil 200 mg ibuprofen” akan kembali Yesterday sebagai TIME_EXPRESSION entitas yang tumpang tindih dengan GENERIC_NAME entitas “ibuprofen.” Namun, itu tidak akan diakui sebagai entitas dalam “kemarin, pasien berjalan anjingnya.”

Tipe

- TIME_TO_MEDICATION_NAME: Tanggal obat diambil. Atribut khusus untuk jenis ini adalah BRAND_NAME dan GENERIC_NAME.
- TIME_TO_DX_NAME: Tanggal kondisi medis terjadi. Atribut untuk jenis ini adalah DX_NAME.
- TIME_TO_TEST_NAME: Tanggal tes dilakukan. Atribut untuk jenis ini adalah TEST_NAME.
- TIME_TO_PROCEDURE_NAME: Tanggal prosedur dilakukan. Atribut untuk jenis ini adalah PROCEDURE_NAME.
- TIME_TO_TREATMENT_NAME: Tanggal pengobatan diberikan. Atribut untuk jenis ini adalah TREATMENT_NAME.

Jenis hubungan

- Hubungan antara entitas dan atribut. Yang diakui Relationship_type adalah sebagai berikut:
Overlap- TIME_EXPRESSION Setuju dengan entitas yang terdeteksi.

Mendeteksi entitas

Note

Jangan gunakan versi DetectEntities operasi ini untuk aplikasi baru. Gunakan versi 2 operasi sebagai gantinya. Semua iterasi dan penyempurnaan fitur baru akan spesifik untuk DetectEntities V2. Untuk informasi selengkapnya, lihat [Mendeteksi entitas \(Versi 2\)](#).

Gunakan DetectEntities operasi untuk mendeteksi entitas medis dalam teks Anda. Mendeteksi entitas dalam kategori berikut:

- ANATOMY
- MEDICAL_CONDITION

- MEDICATION
- PROTECTED_HEALTH_INFORMATION
- TEST_TREATMENT_PROCEDURE

Kelima kategori terdeteksi oleh DetectEntities operasi. Operasi DetectPHI mendeteksi entitas hanya dalam kategori. PROTECTED_HEALTH_INFORMATION Gunakan hanya jika informasi kesehatan yang dilindungi (PHI) diperlukan. Untuk informasi tentang operasi ini, lihat [Mendeteksi PHI](#).

Amazon Comprehend Medical mendeteksi informasi di kelas-kelas berikut:

- Entitas: Referensi tekstual untuk nama objek yang relevan, seperti orang, perawatan, obat-obatan, dan kondisi medis. Misalnya, “Ibuprofen.”
- Kategori: Pengelompokan umum yang dimiliki entitas yang terdeteksi. Misalnya, “Ibuprofen” adalah bagian dari MEDICATION kategori.
- Jenis: Jenis entitas terdeteksi, yang scoped ke kategori. Misalnya, “Ibuprofen” ada dalam GENERIC_NAME tipe MEDICATION kategori.
- Atribut: Informasi yang terkait dengan entitas yang terdeteksi, seperti dosis obat. Misalnya, “200 mg” adalah atribut dari entitas “Ibuprofen”.
- Sifat: Sesuatu yang dipahami Amazon Comprehend Medical tentang suatu entitas, berdasarkan konteks. Misalnya, obat memiliki NEGATION sifat jika pasien tidak meminumnya.

Amazon Comprehend Medical memberi Anda lokasi entitas dalam teks masukan. Di konsol Amazon Comprehend, ini menunjukkan lokasi secara grafis. Bila Anda menggunakan API, itu menunjukkan lokasi dengan offset numerik.

Setiap entitas dan atribut menyertakan skor yang menunjukkan tingkat kepercayaan yang dimiliki Amazon Comprehend Medical dalam keakuratan deteksi. Setiap atribut juga memiliki skor hubungan. Skor ini menunjukkan tingkat kepercayaan yang dimiliki Amazon Comprehend Medical dalam keakuratan hubungan antara atribut dan entitas induknya. Identifikasi ambang batas kepercayaan yang tepat untuk kasus penggunaan Anda. Gunakan ambang batas kepercayaan tinggi dalam situasi yang memerlukan akurasi tinggi, dan filter data yang tidak memenuhi ambang batas Anda.

Kategori anatomi

ANATOMY Kategori mendeteksi referensi ke bagian-bagian tubuh atau sistem tubuh dan lokasi bagian-bagian atau sistem tersebut. Ini berisi dua jenis entitas berikut.

Tipe

- **DIRECTION**: Istilah Directional. Misalnya, kiri, kanan, medial, lateral, atas, bawah, posterior, anterior, distal, proksimal, kontralateral, bilateral, ipsilateral, dorsal, ventral, dan sebagainya.
- **SYSTEM_ORGAN_SITE**: Sistem tubuh, lokasi anatomi atau daerah, dan situs tubuh.

Contoh

Teks “Paru kiri pasien” mengembalikan yang berikut:

Patient's **left** **lung**
• Direction (left) • System organ site (lung)

- “kiri” adalah **DIRECTION** tipe.
- “paru-paru” adalah **SYSTEM_ORGAN_SITE** tipe.

DetectEntitiesOperasi mengembalikan struktur JSON berikut:

```
{
  "Entities": [
    {
      "Id": 0,
      "BeginOffset": 10,
      "EndOffset": 14,
      "Score": 0.9876197576522827,
      "Text": "left",
      "Category": "ANATOMY",
      "Type": "DIRECTION",
      "Traits": []
    },
    {
      "Id": 1,
      "BeginOffset": 15,
      "EndOffset": 19,
      "Score": 0.9820258021354675,
      "Text": "lung",
      "Category": "ANATOMY",
      "Type": "SYSTEM_ORGAN_SITE",
      "Traits": []
    }
  ]
}
```

```
  ],  
  "UnmappedAttributes": []  
}
```

Kategori kondisi medis

MEDICAL_CONDITION Kategori ini mendeteksi gejala dan diagnosis kondisi medis. Ini berisi dua jenis entitas dan tujuh sifat. Satu atau lebih sifat dapat dikaitkan dengan suatu tipe.

Tipe

- **ACUITY**: Penentuan contoh penyakit, seperti kronis, akut, mendadak, persisten, atau bertahap.
- **DX_NAME**: Semua kondisi medis terdaftar. **DX_NAME** Tipe ini termasuk penyakit saat ini, alasan kunjungan, dan riwayat kesehatan.

Atribut

- **QUALITY**: Setiap istilah deskriptif dari kondisi medis, seperti tahap atau kelas.

Ciri-ciri

- **DIAGNOSIS**: Identifikasi kondisi medis yang ditentukan oleh evaluasi gejala. Evaluasi ini berasal dari temuan fisik, laporan laboratorium atau radiologi, atau narasi pasien. Berlaku hanya untuk **DX_NAME** jenisnya.
- **HYPOTHETICAL**: Indikasi bahwa kondisi medis dinyatakan sebagai hipotesis. Berlaku hanya untuk **DX_NAME** jenisnya.
- **LOW_CONFIDENCE**: Indikasi bahwa kondisi medis dinyatakan memiliki ketidakpastian yang tinggi. Ini tidak terkait langsung dengan skor kepercayaan yang diberikan. Berlaku hanya untuk **DX_NAME** jenisnya.
- **NEGATION**: Indikasi bahwa hasil atau tindakan negatif atau tidak dilakukan.
- **PERTAINS_TO_FAMILY**: Indikasi bahwa kondisi medis relevan dengan keluarga pasien, bukan pasien. Berlaku hanya untuk **DX_NAME** jenisnya.
- **SIGN**: Kondisi medis yang dilaporkan dokter. Berlaku hanya untuk **DX_NAME** jenisnya.
- **SYMPTOM**: Kondisi medis yang dilaporkan pasien. Berlaku hanya untuk **DX_NAME** jenisnya.

Contoh

Teks "Pasien menderita sakit kronis 4/10" mengembalikan hal berikut:

Patient is suffering from **chronic** **aching pain** 4/10

Symptom
● Acuity (chronic) ● Dx name (aching pain)

- "sakit sakit" adalah DX_NAME jenisnya.
- SYMPTOM adalah sifat dari tipe "sakit sakit".
- "kronis" adalah ACUITY jenisnya.

DetectEntitiesOperasi mengembalikan struktur JSON berikut:

```
{
  "Entities": [
    {
      "Id": 0,
      "BeginOffset": 26,
      "EndOffset": 33,
      "Score": 0.9961825013160706,
      "Text": "chronic",
      "Category": "MEDICAL_CONDITION",
      "Type": "ACUITY",
      "Traits": []
    },
    {
      "Id": 1,
      "BeginOffset": 34,
      "EndOffset": 45,
      "Score": 0.8380221724510193,
      "Text": "aching pain",
      "Category": "MEDICAL_CONDITION",
      "Type": "DX_NAME",
      "Traits": [
        {
          "Name": "SYMPTOM",
          "Score": 0.6004688739776611
        }
      ]
    }
  ]
}
```

```
    }  
  ],  
  "UnmappedAttributes": []  
}
```

Kategori obat

MEDICATION Kategori mendeteksi informasi pengobatan dan dosis untuk pasien. Ini berisi dua jenis entitas, tujuh atribut, dan satu sifat. Satu atau lebih atribut dapat diterapkan ke tipe.

Tipe

- **BRAND_NAME**: Nama merek obat atau agen terapeutik yang dilindungi hak cipta.
- **GENERIC_NAME**: Nama non-merek, nama bahan, atau campuran formula obat atau agen terapeutik.

Atribut

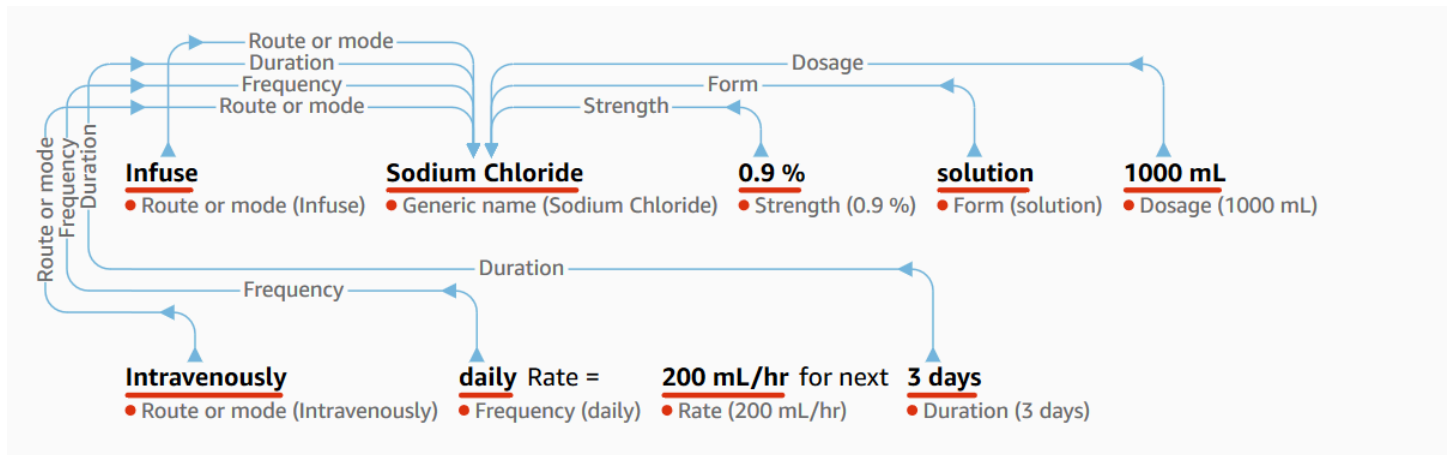
- **DOSAGE**: Jumlah obat yang dipesan.
- **DURATION**: Berapa lama obat harus diberikan.
- **FORM**: Bentuk obatnya.
- **FREQUENCY**: Seberapa sering mengelola obat.
- **RATE**: Tingkat pemberian obat (terutama digunakan untuk infus obat atau infus).
- **ROUTE_OR_MODE**: Metode pemberian obat.
- **STRENGTH**: Kekuatan obat.

Ciri-ciri

- **NEGATION**: Setiap indikasi bahwa pasien tidak minum obat.
- **PAST_HISTORY**: Indikasi bahwa obat yang terdeteksi berasal dari masa lalu pasien (sebelum pertemuan saat ini).

Contoh

Teks “Infuse Sodium Chloride 0,9% solusi 1000 mL intravena harian Tingkat - 200 mL/jam untuk 3 hari ke depan” mengembalikan berikut:



- "Infus" sebagai atribut "ROUTE_OR_MODE" yang terkait dengan tipe "Sodium Chloride".
- "Sodium Chloride" sebagai GENERIC_NAME tipe.
- "0,9%" sebagai STRENGTH atribut yang terkait dengan tipe "Sodium Chloride".
- "solusi" sebagai FORM atribut yang terkait dengan tipe "Sodium Chloride".
- "100 mL sebagai DOSAGE atribut yang terkait dengan tipe "Sodium Chloride".
- "intravena" sebagai ROUTE_OR_MODE atribut yang terkait dengan tipe "Sodium Chloride".
- "harian" sebagai FREQUENCY atribut yang terkait dengan tipe "Sodium Chloride".
- "200 ml/jam" sebagai RATE atribut yang terkait dengan tipe "Sodium Chloride".
- "3 hari berikutnya" sebagai DURATION atribut yang terkait dengan tipe "Sodium Chloride".

DetectEntitiesOperasi mengembalikan struktur JSON berikut:

```
{
  "Entities": [
    {
      "Id": 1,
      "BeginOffset": 7,
      "EndOffset": 22,
      "Score": 0.9998517036437988,
      "Text": "Sodium Chloride",
      "Category": "MEDICATION",
      "Type": "GENERIC_NAME",
      "Traits": [],
      "Attributes": [
        {
          "Type": "ROUTE_OR_MODE",
```



```
    "Score": 0.32359644770622253,  
    "RelationshipScore": 0.9719992280006409,  
    "Id": 0,  
    "BeginOffset": 0,  
    "EndOffset": 6,  
    "Text": "Infuse",  
    "Traits": []  
  },  
  {  
    "Type": "STRENGTH",  
    "Score": 0.9976715445518494,  
    "RelationshipScore": 0.7892051339149475,  
    "Id": 2,  
    "BeginOffset": 23,  
    "EndOffset": 27,  
    "Text": "0.9%",  
    "Traits": []  
  },  
  {  
    "Type": "FORM",  
    "Score": 0.9930835962295532,  
    "RelationshipScore": 0.9956902861595154,  
    "Id": 3,  
    "BeginOffset": 28,  
    "EndOffset": 36,  
    "Text": "solution",  
    "Traits": []  
  },  
  {  
    "Type": "ROUTE_OR_MODE",  
    "Score": 0.9990690350532532,  
    "RelationshipScore": 0.9801701903343201,  
    "Id": 5,  
    "BeginOffset": 45,  
    "EndOffset": 58,  
    "Text": "intravenously",  
    "Traits": []  
  },  
  {  
    "Type": "FREQUENCY",  
    "Score": 0.9539222121238708,  
    "RelationshipScore": 0.9864235520362854,  
    "Id": 6,  
    "BeginOffset": 59,
```

```
        "EndOffset": 64,
        "Text": "daily",
        "Traits": []
    },
    {
        "Type": "DURATION",
        "Score": 0.9392423033714294,
        "RelationshipScore": 0.9961885809898376,
        "Id": 8,
        "BeginOffset": 91,
        "EndOffset": 97,
        "Text": "3 days",
        "Traits": []
    }
]
},
"UnmappedAttributes": [
    {
        "Type": "MEDICATION",
        "Attribute": {
            "Type": "DOSAGE",
            "Score": 0.9922149777412415,
            "Id": 4,
            "BeginOffset": 37,
            "EndOffset": 44,
            "Text": "1000 mL",
            "Traits": []
        }
    },
    {
        "Type": "MEDICATION",
        "Attribute": {
            "Type": "RATE",
            "Score": 0.9728594422340393,
            "Id": 7,
            "BeginOffset": 72,
            "EndOffset": 81,
            "Text": "200 mL/hr",
            "Traits": []
        }
    }
]
```

```
}
```

Kategori Informasi Kesehatan Terlindungi

PROTECTED_HEALTH_INFORMATION Kategori mendeteksi informasi pribadi pasien. Ini berisi delapan jenis entitas. Untuk informasi lengkap tentang PROTECTED_HEALTH_INFORMATION kategori dan bagaimana hal itu terdeteksi, lihat [Mendeteksi PHI](#).

Tipe

- ADDRESS: Semua subdivisi geografis dari alamat fasilitas apa pun, bernama fasilitas medis, atau bangsal dalam fasilitas.
- AGE: Semua komponen usia, rentang usia, atau usia yang disebutkan dalam catatan klinis pasien atau orang lain. Defaultnya adalah dalam beberapa tahun kecuali dinyatakan lain.
- EMAIL: Alamat email apa pun.
- ID: Nomor identifikasi apa pun yang terkait dengan pasien. Ini termasuk nomor spesifik pasien seperti nomor Jaminan Sosial, nomor rekam medis, nomor sertifikat atau lisensi, nomor kendaraan atau perangkat, atau nomor biometrik apa pun. Ini juga termasuk nomor identifikasi fasilitas, nomor uji klinis, tempat perawatan, atau penyedia.
- DATE: Tanggal apa pun yang terkait dengan perawatan pasien atau pasien.
- NAME: Semua nama yang disebutkan dalam catatan klinis. Biasanya, nama-nama milik pasien, keluarga, atau penyedia.
- PHONE_OR_FAX: Telepon, faks, atau nomor pager apa pun. Tidak termasuk nomor telepon bernama, seperti 1-800-QUIT-NOW dan 911.
- PROFESSION: Setiap profesi atau majikan yang disebutkan dalam catatan klinis yang berkaitan dengan pasien atau keluarga pasien. Ini tidak mengacu pada profesi dokter yang disebutkan dalam catatan.

Contoh

Teks *“Pasien adalah John Smith, seorang guru berusia 48 tahun dan penduduk Seattle, Washington.”* mengembalikan berikut:

Patient is **John Smith**, a **48** year old **teacher** and resident
of **Seattle, Washington**.

- Name (John Smith)
- Age (48)
- Profession (teacher)
- Address (Seattle, Washington)

- “John Smith” adalah NAME tipe.
- “48” adalah sebuah AGE tipe.
- “guru” adalah PROFESSION tipe.
- “Seattle, Washington” adalah ADDRESS tipe.

DetectEntitiesOperasi mengembalikan struktur JSON berikut:

```
{
  "Entities": [
    {
      "Id": 0,
      "BeginOffset": 11,
      "EndOffset": 21,
      "Score": 0.9967977404594421,
      "Text": "John Smith",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "NAME",
      "Traits": []
    },
    {
      "Id": 1,
      "BeginOffset": 25,
      "EndOffset": 27,
      "Score": 0.9998422861099243,
      "Text": "48",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "AGE",
      "Traits": []
    },
    {
      "Id": 2,
      "BeginOffset": 37,
      "EndOffset": 44,
      "Score": 0.9079490900039673,
      "Text": "teacher",

```

```
        "Category": "PROTECTED_HEALTH_INFORMATION",
        "Type": "PROFESSION",
        "Traits": []
    },
    {
        "Id": 3,
        "BeginOffset": 61,
        "EndOffset": 80,
        "Score": 0.986108124256134,
        "Text": "Seattle, Washington",
        "Category": "PROTECTED_HEALTH_INFORMATION",
        "Type": "ADDRESS",
        "Traits": []
    }
],
"UnmappedAttributes": []
}
```

Kategori tes, perawatan, dan prosedur

TEST_TREATMENT_PROCEDURE Kategori mendeteksi prosedur yang digunakan untuk menentukan kondisi medis. Ini berisi dua jenis entitas dan dua atribut. Satu atau lebih atribut dapat dikaitkan dengan entitas dari TEST_NAME jenis.

Tipe

- PROCEDURE_NAME: Intervensi sebagai tindakan satu kali dilakukan pada pasien untuk mengobati kondisi medis atau untuk memberikan perawatan pasien.
- TEST_NAME: Prosedur yang dilakukan pada pasien untuk diagnostik, pengukuran, skrining, atau penilaian yang mungkin memiliki nilai yang dihasilkan. Ini termasuk prosedur, evaluasi, atau penilaian apa pun untuk menentukan diagnosis, untuk mengesampingkan suatu kondisi, atau untuk mengukur atau menilai pasien.
- TREATMENT_NAME: Intervensi dilakukan selama rentang waktu untuk memerangi penyakit atau gangguan. Ini termasuk pengelompokan obat, seperti antivirus dan vaksinasi.

Atribut

- TEST_VALUE: Hasil tes. Berlaku hanya untuk jenis TEST_NAME entitas.

- **TEST_UNIT**: Satuan ukuran yang mungkin menyertai nilai tes. Berlaku hanya untuk jenis **TEST_NAME** entitas.

Ciri-ciri

- **FUTURE**: Indikasi bahwa tes, perawatan, atau prosedur mengacu pada tindakan atau peristiwa yang akan terjadi setelah subjek catatan.
- **HYPOTHETICAL**: Indikasi bahwa tes, pengobatan, atau prosedur dinyatakan sebagai hipotesis
- **NEGATION**: Indikasi bahwa tes, pengobatan, atau prosedur negatif atau tidak dilakukan.
- **PAST_HISTORY**: Indikasi bahwa tes, pengobatan, atau prosedur berasal dari masa lalu pasien (sebelum pertemuan saat ini).

Contoh

Teks *USG perut mencatat apendisitis akut, merekomendasikan usus buntu diikuti oleh beberapa seri antibiotik spektrum luas mengembalikan berikut:*

The diagram illustrates the detection of entities in the text: "Abdominal ultrasound noted acute appendicitis, recommend appendectomy followed by several series of broad spectrum antibiotics".

- Abdominal ultrasound** (Test name (Abdominal ultrasound))
- acute** (Acuity (acute))
- appendicitis** (Dx name (appendicitis)) - Labeled as **Diagnosis**
- appendectomy** (Procedure name (appendectomy))
- broad spectrum antibiotics** (Treatment name (broad spectrum antibiotics))

- “Ultrasonografi perut” adalah **TEST_NAME** tipe.
- “akut” adalah **ACUITY** tipe.
- “apendisitis” adalah tipe. **DX_NAME**
- **DIAGNOSIS** adalah sifat tipe “apendisitis”.
- “appendectomy” adalah tipe. **PROCEDURE_NAME**
- “antibiotik spektrum luas” adalah **TREATMENT_NAME** tipe.

DetectEntities Operasi mengembalikan struktur JSON berikut:

```
{
  "Entities": [
    {
```

```
    "Id": 0,
    "BeginOffset": 0,
    "EndOffset": 20,
    "Score": 0.94855135679245,
    "Text": "Abdominal ultrasound",
    "Category": "TEST_TREATMENT_PROCEDURE",
    "Type": "TEST_NAME",
    "Traits": []
  },
  {
    "Id": 3,
    "BeginOffset": 27,
    "EndOffset": 32,
    "Score": 0.9067845940589905,
    "Text": "acute",
    "Category": "MEDICAL_CONDITION",
    "Type": "ACUITY",
    "Traits": []
  },
  {
    "Id": 4,
    "BeginOffset": 33,
    "EndOffset": 45,
    "Score": 0.9954161643981934,
    "Text": "appendicitis",
    "Category": "MEDICAL_CONDITION",
    "Type": "DX_NAME",
    "Traits": [
      {
        "Name": "DIAGNOSIS",
        "Score": 0.9528769254684448
      }
    ]
  },
  {
    "Id": 1,
    "BeginOffset": 57,
    "EndOffset": 69,
    "Score": 0.9957893490791321,
    "Text": "appendectomy",
    "Category": "TEST_TREATMENT_PROCEDURE",
    "Type": "PROCEDURE_NAME",
    "Traits": []
  },
}
```

```
{
  "Id": 2,
  "BeginOffset": 100,
  "EndOffset": 126,
  "Score": 0.9437107443809509,
  "Text": "broad spectrum antibiotics",
  "Category": "TEST_TREATMENT_PROCEDURE",
  "Type": "TREATMENT_NAME",
  "Traits": []
},
"UnmappedAttributes": []
}
```

Mendeteksi PHI

Gunakan operasi DetectPhi ketika Anda hanya ingin mendeteksi data Informasi Kesehatan yang Dilindungi (PHI) saat memindai teks klinis. Untuk mendeteksi semua entitas yang tersedia dalam teks klinis gunakan DetectEntitiesV2.

API ini paling baik untuk kasus penggunaan di mana hanya mendeteksi entitas PHI yang diperlukan. Untuk informasi tentang informasi dalam kategori non-PHI, lihat. [Mendeteksi entitas](#)

Important

Amazon Comprehend Medical memberikan skor kepercayaan yang menunjukkan tingkat kepercayaan pada keakuratan entitas yang terdeteksi. Evaluasi skor kepercayaan ini dan identifikasi ambang kepercayaan yang tepat untuk kasus penggunaan Anda. Untuk kasus penggunaan kepatuhan tertentu, kami menyarankan Anda menggunakan tinjauan manusia tambahan atau metode lain untuk mengonfirmasi keakuratan PHI yang terdeteksi.

Di bawah undang-undang HIPAA, PHI yang didasarkan pada daftar 18 pengidentifikasi harus diperlakukan dengan hati-hati. Amazon Comprehend Medical mendeteksi entitas yang terkait dengan pengidentifikasi ini tetapi entitas ini tidak memetakan 1:1 ke daftar yang ditentukan oleh metode Safe Harbor. Tidak semua pengidentifikasi terkandung dalam teks klinis yang tidak terstruktur, tetapi Amazon Comprehend Medical mencakup semua pengidentifikasi yang relevan. Pengidentifikasi ini terdiri dari data yang dapat digunakan untuk mengidentifikasi pasien individu, termasuk daftar berikut.

Untuk informasi lebih lanjut, lihat [Privasi Informasi Kesehatan](#) di situs web Kesehatan dan Layanan Kemanusiaan Pemerintah AS.

Setiap entitas terkait Phi menyertakan skor (Score dalam respons) yang menunjukkan tingkat kepercayaan yang dimiliki Amazon Comprehend Medical dalam keakuratan deteksi. Identifikasi ambang kepercayaan yang tepat untuk kasus penggunaan Anda dan saring entitas yang tidak memenuhinya. Saat mengidentifikasi kejadian PHI, mungkin lebih baik menggunakan ambang kepercayaan rendah untuk penyaringan guna menangkap lebih banyak entitas potensial yang terdeteksi. Ini terutama benar ketika tidak menggunakan nilai entitas yang terdeteksi dalam kasus penggunaan kepatuhan.

Entitas terkait Phi berikut dapat dideteksi dengan menjalankan operasi DetectPhi atau V2: DetectEntities

Entitas PHI yang Terdeteksi

Entitas	Deskripsi	Kategori HIPAA
USIA	Semua komponen usia, rentang usia, dan usia apa pun yang disebutkan, baik itu pasien atau anggota keluarga atau orang lain yang terlibat dalam catatan. Default dalam beberapa tahun kecuali dinyatakan lain.	3. Tanggal yang terkait dengan individu
TANGGAL	Tanggal apa pun yang terkait dengan perawatan pasien atau pasien.	3. Tanggal yang terkait dengan individu
NAME	Semua nama yang disebutkan dalam catatan klinis, biasanya milik pasien, keluarga, atau penyedia.	1. Nama
PHONE_OR_FAX	Telepon, faks, pager apa pun; tidak termasuk nomor telepon	4. Nomor telepon 5. Nomor FAX

Entitas	Deskripsi	Kategori HIPAA
	bernama seperti 1-800-QUIT-NOW serta 911.	
Email	Alamat email apa pun.	6. Alamat email
ID	Segala jenis nomor yang terkait dengan identitas pasien. Ini termasuk nomor jaminan sosial mereka, nomor rekam medis, nomor identifikasi fasilitas, nomor uji klinis, sertifikat atau nomor lisensi, nomor kendaraan atau perangkat. Ini juga termasuk nomor biometrik, dan angka yang mengidentifikasi tempat perawatan atau penyedia.	<ul style="list-style-type: none"> 7. Nomor Jaminan Sosial 8. Nomor Rekam Medis 9. Nomor Rencana Kesehatan 10. Nomor akun 11. Nomor Sertifikat/Lisensi 12. Pengidentifikasi kendaraan 13. Nomor perangkat 16. Informasi biometrik 18. Karakteristik pengenalan lainnya
URL	URL web apa pun.	14. URL
MENEGUR	Ini termasuk semua subdivisi geografis dari alamat fasilitas apa pun, fasilitas medis bernama, atau bangsal dalam fasilitas.	2. Lokasi geografis
PROFESI	Termasuk profesi atau majikan yang disebutkan dalam catatan yang berkaitan dengan pasien atau keluarga pasien.	18. Karakteristik pengenalan lainnya

Contoh

Teks “Pasien adalah John Smith, seorang guru berusia 48 tahun dan penduduk Seattle, Washington.” mengembalikan:

- “John Smith” sebagai entitas tipe NAME dalam PROTECTED_HEALTH_INFORMATION kategori.
- “48” sebagai entitas tipe AGE dalam PROTECTED_HEALTH_INFORMATION kategori.
- “guru” sebagai entitas tipe PROFESSION (mengidentifikasi karakteristik) dalam PROTECTED_HEALTH_INFORMATION kategori.
- “Seattle, Washington” sebagai ADDRESS entitas dalam PROTECTED_HEALTH_INFORMATION kategori.

Di konsol Amazon Comprehend Medical, ini ditampilkan seperti ini:

```
Patient is John Smith, a 48 year old teacher and resident
           ● Name (John Smith) ● Age (48) ● Profession (teacher)

of Seattle, Washington.
   ● Address (Seattle , Washington)
```

Saat menggunakan operasi DetectPhi, responsnya muncul seperti ini. Saat Anda menggunakan DetectionJob operasi startPhi, Amazon Comprehend Medical membuat file di lokasi keluaran dengan struktur ini.

```
{
  "Entities": [
    {
      "Id": 0,
      "BeginOffset": 11,
      "EndOffset": 21,
      "Score": 0.997368335723877,
      "Text": "John Smith",
      "Category": "PROTECTED_HEALTH_INFORMATION",
      "Type": "NAME",
      "Traits": []
    },
    {
      "Id": 1,
      "BeginOffset": 25,
      "EndOffset": 27,
```

```
    "Score": 0.9998362064361572,
    "Text": "48",
    "Category": "PROTECTED_HEALTH_INFORMATION",
    "Type": "AGE",
    "Traits": []
  },
  {
    "Id": 2,
    "BeginOffset": 37,
    "EndOffset": 44,
    "Score": 0.8661606311798096,
    "Text": "teacher",
    "Category": "PROTECTED_HEALTH_INFORMATION",
    "Type": "PROFESSION",
    "Traits": []
  },
  {
    "Id": 3,
    "BeginOffset": 61,
    "EndOffset": 68,
    "Score": 0.9629441499710083,
    "Text": "Seattle",
    "Category": "PROTECTED_HEALTH_INFORMATION",
    "Type": "ADDRESS",
    "Traits": []
  },
  {
    "Id": 4,
    "BeginOffset": 78,
    "EndOffset": 88,
    "Score": 0.38217034935951233,
    "Text": "Washington",
    "Category": "PROTECTED_HEALTH_INFORMATION",
    "Type": "ADDRESS",
    "Traits": []
  }
],
"UnmappedAttributes": []
}
```

API batch analisis teks

Gunakan Amazon Comprehend Medical untuk menganalisis teks medis yang disimpan dalam ember Amazon S3. Analisis hingga 10 GB dokumen dalam satu batch. Anda menggunakan konsol untuk membuat dan mengelola pekerjaan analisis batch, atau menggunakan API batch untuk mendeteksi entitas medis, termasuk informasi kesehatan yang dilindungi (PHI). API memulai, menghentikan, mencantumkan, dan menjelaskan pekerjaan analisis batch yang sedang berlangsung.

[Informasi harga untuk analisis batch dan operasi Amazon Comprehend Medical lainnya dapat ditemukan di sini.](#)

Pemberitahuan penting

Operasi analisis batch Amazon Comprehend Medical bukanlah pengganti saran medis profesional, diagnosis, atau perawatan. Identifikasi ambang kepercayaan yang tepat untuk kasus penggunaan Anda, dan gunakan ambang kepercayaan tinggi dalam situasi yang membutuhkan akurasi tinggi. Untuk kasus penggunaan tertentu, hasil harus ditinjau dan diverifikasi oleh pengulas manusia yang terlatih dengan tepat. Semua operasi Amazon Comprehend Medical hanya boleh digunakan dalam skenario perawatan pasien setelah ditinjau untuk akurasi dan penilaian medis yang baik oleh profesional medis terlatih.

Melakukan analisis batch menggunakan API

Anda dapat menjalankan pekerjaan analisis batch menggunakan konsol Amazon Comprehend Medical atau Amazon Comprehend Medical Batch API.

Prasyarat

Saat Anda menggunakan Amazon Comprehend Medical API, buat kebijakan AWS Identity Access and Management (IAM) dan lampirkan ke peran IAM. Untuk mempelajari lebih lanjut tentang peran IAM dan kebijakan kepercayaan, lihat Kebijakan [dan Izin IAM](#).

1. Unggah data Anda ke dalam bucket S3.
2. Untuk memulai pekerjaan analisis baru, gunakan operasi `StartEntitiesDetection V2Job` atau operasi `DetectionJob StartPhi`. Saat Anda memulai pekerjaan, beri tahu Amazon Comprehend Medical nama bucket input S3 yang berisi file input dan tentukan bucket S3 keluaran untuk menulis file setelah analisis batch.

3. Pantau kemajuan pekerjaan dengan menggunakan konsol atau operasi `DescribeEntitiesDetection V2Job` atau operasi `DetectionJob DescribePhi`. Selain itu, `ListEntitiesDetection V2Jobs` dan `ListPhi DetectionJobs` memungkinkan Anda untuk melihat status semua ontologi yang menghubungkan pekerjaan analisis batch.
4. Jika Anda perlu menghentikan pekerjaan yang sedang berlangsung, gunakan `StopEntitiesDetection V2Job` atau `StopPhi DetectionJob` untuk menghentikan analisis.
5. Untuk melihat hasil pekerjaan analisis Anda, lihat bucket S3 keluaran yang Anda konfigurasi saat memulai pekerjaan.

Melakukan analisis batch menggunakan konsol

1. Unggah data Anda ke dalam bucket S3.
2. Untuk memulai pekerjaan analisis baru, pilih jenis analisis yang akan Anda lakukan. Kemudian berikan nama bucket S3 yang berisi file input dan nama bucket S3 tempat Anda ingin mengirim file output.
3. Pantau status pekerjaan Anda saat sedang berlangsung. Dari konsol, Anda dapat melihat semua operasi analisis batch dan statusnya, termasuk kapan analisis dimulai dan berakhir.
4. Untuk melihat hasil pekerjaan analisis Anda, lihat bucket keluaran S3 yang Anda konfigurasi saat memulai pekerjaan.

Kebijakan IAM untuk operasi batch

Peran IAM yang memanggil API batch Amazon Comprehend Medical harus memiliki kebijakan yang memberikan akses ke bucket S3 yang berisi file input dan output. Itu juga harus diberi hubungan kepercayaan yang memungkinkan layanan Amazon Comprehend Medical untuk mengambil peran. Untuk mempelajari lebih lanjut tentang peran IAM dan kebijakan kepercayaan, lihat Peran [IAM](#).

Peran harus memiliki kebijakan berikut.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject"
```

```

    ],
    "Resource": [
      "arn:aws:s3:::input-bucket/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:ListBucket"
    ],
    "Resource": [
      "arn:aws:s3:::input-bucket",
      "arn:aws:s3:::output-bucket",
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:PutObject"
    ],
    "Resource": [
      "arn:aws:s3:::output-bucket/*"
    ],
    "Effect": "Allow"
  }
]
}

```

Peran tersebut harus memiliki hubungan kepercayaan berikut. Disarankan agar Anda menggunakan kunci `aws:SourceAccount` dan `aws:SourceArn` kondisi untuk mencegah masalah keamanan wakil yang membingungkan. Untuk mempelajari lebih lanjut tentang masalah wakil yang bingung dan cara melindungi AWS akun Anda, lihat [Masalah wakil yang bingung](#) dalam dokumentasi IAM.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {

```

```

    "Service": [
      "comprehendmedical.amazonaws.com"
    ],
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "account_id"
      },
      "ArnLike": {
        "aws:SourceArn": [
          "arn:aws:s3:::input-bucket/*",
          "arn:aws:s3:::output-bucket"
        ]
      }
    }
  }
}

```

File keluaran analisis Batch

Amazon Comprehend Medical membuat satu file output untuk setiap file input dalam batch. File memiliki ekstensi `.out`. Amazon Comprehend Medical pertama-tama membuat direktori di bucket output S3 menggunakan `AwsAccountIdJobType- JobId` sebagai namanya, dan kemudian menulis semua file output untuk batch ke direktori ini. Amazon Comprehend Medical membuat direktori baru ini sehingga output dari satu pekerjaan tidak menimpa output yang lain.

Output dari operasi batch menghasilkan output yang sama dengan operasi sinkron. Untuk contoh output yang dihasilkan oleh Amazon Comprehend Medical, lihat [Mendeteksi entitas](#)

Setiap operasi batch menghasilkan tiga file manifes yang berisi informasi tentang pekerjaan.

- **Manifest**— Meringkas pekerjaan. Memberikan informasi tentang parameter yang digunakan untuk pekerjaan, ukuran total pekerjaan, dan jumlah file yang diproses.
- **success**— Memberikan informasi tentang file yang berhasil diproses. Termasuk nama file input dan output dan ukuran file input.
- **unprocessed**— Daftar file yang tidak diproses oleh pekerjaan batch. Biasanya ini karena file ditambahkan ke direktori input setelah pekerjaan batch dimulai.

Amazon Comprehend Medical menulis file ke direktori output yang Anda tentukan untuk pekerjaan batch. Bagian berikut menunjukkan struktur file manifes.

File manifes batch

Berikut ini adalah struktur JSON dari file manifes batch.

```
{
  "Summary" : {
    "Status" : "COMPLETED | FAILED | PARTIAL_SUCCESS | STOPPED",
    "JobType" : "DetectEntitiesJob | PHIDetection",
    "InputDataConfiguration" : {
      "Bucket" : "input bucket",
      "Path" : "path to files/account ID-job type-job ID"
    },
    "OutputDataConfiguration" : {
      "Bucket" : "output bucket",
      "Path" : "path to files"
    },
    "InputFileCount" : number of files in input bucket,
    "TotalMeteredCharacters" : total characters processed from all files,
    "UnprocessedFilesCount" : number of files not processed,
    "SuccessFilesCount" : total number of files processed,
    "TotalDurationSeconds" : time required for processing,
    "SuccessfulFilesListLocation" : "path to file",
    "UnprocessedFilesListLocation" : "path to file"
  }
}
```

File manifes sukses

Berikut ini adalah struktur JSON dari file yang berisi informasi tentang file yang berhasil diproses.

```
{
  "Files": [{
    "Input": "input path/input file name",
    "Output": "output path/output file name",
    "InputSize": size in bytes of input file
  }, {
    "Input": "input path/input file name",
    "Output": "output path/output file name",
    "InputSize": size in bytes of input file
  }
]
```

```
    }]  
}
```

File manifes yang belum diproses

Berikut ini adalah struktur JSON dari file manifes yang berisi informasi tentang file yang belum diproses.

```
{  
  "Files": [  
    "input path/input file name",  
    "input path/input file name"  
  ]  
}
```

Ontologi menghubungkan

Gunakan Amazon Comprehend Medical untuk mendeteksi entitas dalam teks klinis dan menghubungkan entitas tersebut dengan konsep dalam ontologi medis standar, termasuk RxNorm basis pengetahuan, ICD-10-CM, dan SNOMED CT. Anda dapat melakukan analisis baik pada file tunggal atau sebagai analisis batch pada dokumen besar atau beberapa file yang disimpan dalam Amazon Simple Storage Service (S3).

Penautan ICD-10-CM

Gunakan InfericD10CM untuk mendeteksi kemungkinan kondisi medis sebagai entitas dan menautkannya ke kode dari [Klasifikasi Penyakit Internasional versi 2022, Revisi ke-10, Modifikasi Klinis \(ICD-10-CM\)](#). ICD-10-CM disediakan oleh Pusat Pengendalian dan Pencegahan Penyakit AS (CDC).

Ketika kondisi medis terdeteksi, `InferICD10CM` kembalikan kode dan deskripsi ICD-10-CM yang cocok. Kondisi yang terdeteksi tercantum dalam urutan kepercayaan menurun. Skor menunjukkan kepercayaan pada keakuratan entitas yang cocok dengan konsep yang ditemukan dalam teks. Informasi terkait seperti riwayat keluarga, tanda, gejala, dan negasi diakui sebagai sifat. Informasi tambahan seperti penunjukan anatomi dan ketajaman terdaftar sebagai atribut.

`InfericD10cm` sangat cocok untuk skenario berikut:

- Membantu pengkodean medis profesional dari catatan pasien
- Studi klinis dan uji coba
- Integrasi dengan sistem perangkat lunak medis
- Deteksi dan diagnosis dini
- Manajemen kesehatan penduduk

Kategori ICD-10-CM

`InfericD10cm` mendeteksi entitas dalam kategori. `MEDICAL_CONDITION` Informasi terkait tambahan juga terdeteksi dan dihubungkan sebagai atribut atau sifat.

Jenis ICD-10-CM

`InfericD10cm` mendeteksi entitas dari jenis dan. `DX_NAME` `TIME_EXPRESSION`

Ciri-ciri ICD-10-CM

InfericD10cm mendeteksi informasi kontekstual berikut sebagai sifat:

- **DIAGNOSIS**: Identifikasi kondisi medis yang ditentukan oleh evaluasi gejala.
- **HYPOTHETICAL**: Indikasi bahwa suatu kondisi medis dinyatakan sebagai hipotesis.
- **LOW_CONFIDENCE**: Indikasi bahwa suatu kondisi medis dinyatakan memiliki ketidakpastian yang tinggi. Ini tidak terkait langsung dengan skor kepercayaan yang diberikan.
- **NEGATION**: Indikasi bahwa kondisi medis tidak ada.
- **PERTAINS_TO_FAMILY**: Indikasi bahwa suatu kondisi medis relevan dengan keluarga pasien, bukan pasien.
- **SIGN**: Kondisi medis yang dilaporkan oleh dokter.
- **SYMPTOM**: Kondisi medis yang dilaporkan oleh pasien.

Atribut ICD-10-CM

InfericD10cm mendeteksi informasi kontekstual berikut sebagai atribut:

- **DIRECTION**: Istilah terarah. Misalnya, kiri, kanan, medial, lateral, atas, bawah, posterior, anterior, distal, proksimal, kontralateral, bilateral, ipsilateral, dorsal, atau ventral.
- **SYSTEM_ORGAN_SITE**: Lokasi anatomi.
- **ACUITY**: Penentuan contoh penyakit, seperti kronis, akut, tiba-tiba, persisten, atau bertahap. Ini hanya berlaku untuk **MEDICAL_CONDITION** jenisnya.
- **QUALITY**: Setiap istilah deskriptif dari kondisi medis, seperti tahap atau kelas.

Kategori ekspresi waktu

TIME_EXPRESSION Kategori mendeteksi entitas yang terkait dengan waktu. Ini termasuk entitas seperti tanggal, dan ekspresi waktu seperti “tiga hari yang lalu,” “hari ini,” “saat ini,” “hari masuk,” “bulan lalu,” atau “16 hari.” Hasil dalam kategori ini hanya dikembalikan jika dikaitkan dengan entitas. Misalnya, ungkapan, “Kemarin, pasien didiagnosis dengan influenza” akan kembali Yesterday sebagai **TIME_EXPRESSION** entitas yang tumpang tindih dengan **DX_NAME** entitas, “influenza.” Namun, “kemarin” tidak akan diakui sebagai entitas dalam ungkapan, “kemarin, pasien berjalan dengan anjingnya.”

Tipe

Jenis yang diakui TIME_EXPRESSION adalah TIME_TO_DX_NAME: tanggal terjadinya kondisi medis. Atribut untuk tipe ini adalah DX_NAME.

Jenis hubungan

RELATIONSHIP_TYPE Ini mengacu pada hubungan antara entitas dan atribut. Yang diakui RELATIONSHIP_TYPE adalah OVERLAP — TIME_EXPRESSION kesepakatan dengan entitas yang terdeteksi.

Contoh masukan dan respons

Contoh berikut menunjukkan bagaimana InferICD10CM operasi bekerja. Berikut ini adalah contoh teks masukan.

“Pasien adalah pasien wanita berusia 71 tahun dari Dr. X. Pasien yang dipresentasikan ke ruang gawat darurat tadi malam dengan riwayat nyeri perut sekitar 7 hingga 8 hari yang telah persisten. Dia tidak mengalami mual dan muntah, tetapi memiliki anoreksia terkait persisten. Dia mengalami flatus, tetapi memiliki beberapa gejala obstipasi dengan buang air besar terakhir dua hari yang lalu. Dia menyangkal darah merah cerah per rektum dan tidak ada riwayat melena baru-baru ini. Kolonoskopi terakhirnya adalah sekitar 5 tahun yang lalu dengan Dr. Y. Dia tidak memiliki demam atau menggigil yang pasti dan tidak ada riwayat penyakit kuning. Pasien menyangkal adanya penurunan berat badan yang signifikan baru-baru ini.”

Untuk teks input, operasi InferICD10cm mengembalikan output berikut (disingkat singkatnya).

```
{
  "Entities": [
    {
      "Id": 1,
      "Text": "abdominal pain",
      "Category": "MEDICAL_CONDITION",
      "Type": "DX_NAME",
      "Score": 0.9606665968894958,
      "BeginOffset": 153,
      "EndOffset": 167,
      "Attributes": [
        {
```

```
        "Type": "ACUITY",
        "Score": 0.764342725276947,
        "RelationshipScore": 0.9999940395355225,
        "Id": 2,
        "BeginOffset": 183,
        "EndOffset": 193,
        "Text": "persistent",
        "Traits": []
    }
],
"Traits": [
    {
        "Name": "SYMPTOM",
        "Score": 0.7559975981712341
    }
],
"ICD10CMConcepts": [
    {
        "Description": "Unspecified abdominal pain",
        "Code": "R10.9",
        "Score": 0.7775180339813232
    },
    {
        "Description": "Epigastric pain",
        "Code": "R10.13",
        "Score": 0.6876822710037231
    },
    {
        "Description": "Lower abdominal pain, unspecified",
        "Code": "R10.30",
        "Score": 0.6758853197097778
    },
    {
        "Description": "Generalized abdominal pain",
        "Code": "R10.84",
        "Score": 0.6746202707290649
    },
    {
        "Description": "Upper abdominal pain, unspecified",
        "Code": "R10.10",
        "Score": 0.6702126860618591
    }
]
}
```

```
...
  "ModelVersion": "2.5.0.20220401"
}
```

InfericD10cm juga mengenali ketika suatu entitas dinegasikan dalam teks. Misalnya, jika pasien tidak mengalami gejala, baik gejala maupun negasi diidentifikasi sebagai sifat dan terdaftar dengan skor kepercayaan. Berdasarkan masukan untuk contoh sebelumnya, gejala Nausea akan tercantum di bawah NEGATION karena pasien tidak mengalami mual.

```
{
  "Id": 3,
  "Text": "nausea",
  "Category": "MEDICAL_CONDITION",
  "Type": "DX_NAME",
  "Score": 0.9962648749351501,
  "BeginOffset": 210,
  "EndOffset": 216,
  "Attributes": [],
  "Traits": [
    {
      "Name": "SYMPTOM",
      "Score": 0.9296342730522156
    },
    {
      "Name": "NEGATION",
      "Score": 0.9620923399925232
    }
  ],
  "ICD10CMConcepts": [
    {
      "Description": "Nausea with vomiting, unspecified",
      "Code": "R11.2",
      "Score": 0.8000147938728333
    },
    {
      "Description": "Nausea",
      "Code": "R11.0",
      "Score": 0.7653312683105469
    }
  ]
}
```

RxNorm menghubungkan

Gunakan InferRxNormoperasi untuk mengidentifikasi obat yang tercantum dalam catatan pasien sebagai entitas. Operasi ini juga menghubungkan entitas tersebut ke pengidentifikasi konsep (RxCui) dari [RxNorm database dari Perpustakaan Kedokteran Nasional](#). Sumber untuk setiap RxCui adalah RxNorm 2022-11-07 dan Rilis. RxTerms Setiap RxCui unik untuk kekuatan dan bentuk dosis yang berbeda. Amazon Comprehend Medical mencantumkan RXCUIS yang berpotensi cocok untuk setiap obat yang dideteksi dalam urutan menurun berdasarkan skor kepercayaan. Gunakan kode RxCui untuk analisis hilir yang tidak mungkin dilakukan dengan teks tidak terstruktur. Informasi terkait seperti kekuatan, frekuensi, dosis, bentuk dosis, dan rute pemberian terdaftar sebagai atribut dalam format JSON.

Anda dapat menggunakan InferRxNormuntuk skenario berikut:

- Skrining untuk obat yang telah diminum pasien.
- Mencegah reaksi negatif yang berpotensi antara obat yang baru diresepkan dan obat yang sedang dikonsumsi pasien.
- Skrining untuk dimasukkan dalam uji klinis berdasarkan riwayat obat menggunakan RxCui.
- Memeriksa apakah dosis dan frekuensi obat sesuai.
- Skrining untuk penggunaan, indikasi, dan efek samping obat.
- Mengelola kesehatan penduduk.

Pemberitahuan penting

InferRxNormOperasi Amazon Comprehend Medical bukanlah pengganti saran medis profesional, diagnosis, atau perawatan. Identifikasi ambang kepercayaan yang tepat untuk kasus penggunaan Anda, dan gunakan ambang kepercayaan tinggi dalam situasi yang membutuhkan akurasi tinggi. Hanya gunakan operasi Amazon Comprehend Medical dalam skenario perawatan pasien setelah meninjau keakuratan dan menerima penilaian yang baik oleh profesional medis terlatih.

RxNorm kategori

InferRxNormmendeteksi entitas dalam MEDICATION kategori. Ini juga mendeteksi informasi terkait tambahan yang ditautkan sebagai atribut atau sifat.

RxNorm jenis

Jenis-jenis entitas dalam Medication kategori tersebut adalah

- BRAND_NAME: Nama merek obat atau agen terapeutik yang dilindungi hak cipta.
- GENERIC_NAME: Nama non-merek, nama bahan, atau campuran formula obat atau agen terapeutik.

RxNorm atribut

- DOSAGE: Jumlah obat yang dipesan.
- DURATION: Berapa lama obat harus diberikan.
- FORM: Bentuk obatnya.
- FREQUENCY: Seberapa sering memberikan obat.
- RATE: Tingkat pemberian obat (terutama untuk infus obat atau infus).
- ROUTE_OR_MODE: Metode pemberian obat.
- STRENGTH: Kekuatan obat.

RxNorm sifat

- NEGATION: Setiap indikasi bahwa pasien tidak minum obat.
- PAST_HISTORY: Indikasi bahwa obat yang terdeteksi berasal dari masa lalu pasien (sebelum pertemuan saat ini).

Contoh masukan dan respons

Bagian ini menunjukkan contoh penggunaan InferRxNorm operasi menggunakan file AWS Command Line Interface. Contoh kode ini menggunakan catatan medis berikut sebagai masukan: fluoride topikal (fluoride 1,1% gel topikal) 1 aplikasi Topikal setiap hari Sikat gigi sebelum tidur, ludah, jangan bilas, makan atau minum selama 20-30 menit.

```
aws comprehendmedical infer-rx-norm --text 'fluoride topikal ( fluoride 1.1 % topical gel ) 1 application Topically daily Brush onto teeth before bed time , spit , do not rinse, eat or drink for 20-30 minutes'
```

Berdasarkan masukan sebelumnya, Anda menerima respons JSON berikut:

```
{
  "Entities": [
    {
      "Id": 1,
      "Text": "fluoride",
      "Category": "MEDICATION",
      "Type": "GENERIC_NAME",
      "Score": 0.9765048027038574,
      "BeginOffset": 19,
      "EndOffset": 27,
      "Attributes": [],
      "Traits": [],
      "RxNormConcepts": [
        {
          "Description": "fluorine",
          "Code": "1310123",
          "Score": 0.9384168982505798
        },
        {
          "Description": "sodium fluoride",
          "Code": "9873",
          "Score": 0.9174549579620361
        },
        {
          "Description": "magnesium fluoride",
          "Code": "1435860",
          "Score": 0.8124921917915344
        },
        {
          "Description": "sulfuryl fluoride",
          "Code": "2289224",
          "Score": 0.5632417798042297
        },
        {
          "Description": "acidulated phosphate fluoride",
          "Code": "236",
          "Score": 0.41748538613319397
        }
      ]
    }
  ],
  "ModelVersion": "2.2.0.20221003"
}
```

```
}
```

`infer-rx-norm` Operasi mengenali sifat negasi juga.

```
aws comprehendmedical infer-rx-norm --text 'patient is not on warfarin'
```

Berdasarkan masukan sebelumnya, Anda menerima respons JSON berikut:

```
{
  "Entities": [
    {
      "Id": 1,
      "Text": "warfarin",
      "Category": "MEDICATION",
      "Type": "GENERIC_NAME",
      "Score": 0.9970192909240723,
      "BeginOffset": 18,
      "EndOffset": 26,
      "Attributes": [],
      "Traits": [
        {
          "Name": "NEGATION",
          "Score": 0.8079015016555786
        }
      ]
    },
    "RxNormConcepts": [
      {
        "Description": "warfarin",
        "Code": "11289",
        "Score": 0.9439865350723267
      },
      {
        "Description": "warfarin sodium 2 MG Oral Tablet",
        "Code": "855302",
        "Score": 0.5045595169067383
      },
      {
        "Description": "warfarin sodium 10 MG Oral Tablet",
        "Code": "855296",
        "Score": 0.40246912837028503
      },
      {
        "Description": "warfarin sodium 2 MG Oral Tablet [Coumadin]",
```

```
        "Code": "855304",
        "Score": 0.22325271368026733
    },
    {
        "Description": "warfarin sodium 10 MG Oral Tablet [Jantoven]",
        "Code": "855300",
        "Score": 0.13163453340530396
    }
]
},
"ModelVersion": "2.2.0.20221003"
}
```

Menghubungkan CT SNOMED

Gunakan `InfersNomeDCT` untuk mendeteksi entitas medis dan menghubungkannya dengan konsep dari versi 2022-03 dari Nomenklatur Kedokteran Sistematisasi, Istilah Klinis (SNOMED CT). SNOMED CT memberi Anda kosakata konsep medis yang komprehensif, termasuk kondisi medis dan anatomi, tes medis, perawatan, dan prosedur. Untuk mempelajari lebih lanjut tentang SNOMED CT, kunjungi [SNOMED CT](#).

Untuk setiap entitas medis yang terdeteksi, Amazon Comprehend Medical mencantumkan lima ID konsep CT SNOMED teratas dan deskripsi yang terkait dengan konsep medis, bersama dengan skor kepercayaan untuk menunjukkan kepercayaan model dalam prediksinya. ID konsep CT SNOMED terdaftar dalam urutan kepercayaan menurun bersama dengan skor kepercayaan. ID konsep CT SNOMED kemudian dapat digunakan untuk menyusun data klinis pasien untuk pengkodean medis, pelaporan, atau analitik klinis saat Anda menggunakannya dengan hierarki poli-SNOMED CT.

`InfersNoMEDCT` tersedia untuk pelanggan di AS. [Untuk informasi tentang SNOMED CT di negara lain, dan informasi tentang lisensi SNOMED CT, lihat SNOMED CT.](#)

`InfersNoMEDCT` sangat cocok untuk skenario berikut:

- Bantuan untuk pengkodean medis profesional dalam catatan pasien
- Studi klinis dan uji coba
- Manajemen kesehatan penduduk

InfersNomeDCT mendeteksi entitas dalam kategori berikut. Informasi kontekstual tambahan juga terdeteksi dan dihubungkan sebagai atribut atau sifat.

- **MEDICAL_CONDITION**: Tanda, gejala, dan diagnosis kondisi medis.
- **ANATOMY**: Bagian-bagian tubuh atau sistem tubuh dan lokasi bagian atau sistem tersebut.
- **TEST_TREATMENT_PROCEDURE**: Prosedur yang digunakan untuk menentukan kondisi medis.

Kategori anatomi

ANATOMY Kategori ini mendeteksi referensi ke bagian-bagian tubuh atau sistem tubuh dan lokasi bagian atau sistem tersebut.

Atribut

Atribut berikut terdeteksi untuk **ANATOMY** kategori:

- **DIRECTION**: Istilah terarah. Misalnya, kiri, kanan, medial, lateral, atas, bawah, posterior, anterior, distal, proksimal, kontralateral, bilateral, ipsilateral, dorsal, atau ventral.
- **SYSTEM_ORGAN_SITE**: Sistem tubuh, lokasi atau daerah anatomi, dan situs tubuh.

Kategori kondisi medis

MEDICAL_CONDITION Kategori ini mendeteksi tanda, gejala, dan diagnosis kondisi medis.

Tipe

Untuk kategori **MEDICAL_CONDITION**, jenis berikut terdeteksi:

- **DX_NAME**: Identifikasi kondisi medis yang ditentukan oleh evaluasi gejala.

Atribut

Atribut berikut terdeteksi untuk **MEDICAL_CONDITION** kategori:

- **ACUITY**: Penentuan contoh penyakit, seperti kronis, akut, tiba-tiba, persisten, atau bertahap.
- **QUALITY**: Setiap istilah deskriptif dari kondisi medis, seperti tahap atau kelas.
- **DIRECTION**: Istilah terarah. Misalnya, kiri, kanan medial, lateral, atas, bawah, posterior, anterior, distal, proksimal, kontralateral, bilateral, ipsilateral, dorsal, atau ventral.

- **SYSTEM_ORGAN_SITE**: Sistem tubuh, lokasi atau daerah anatomi, dan situs tubuh.

Sifat-sifat

Ciri-ciri berikut terdeteksi untuk **MEDICAL_CONDITION** kategori:

- **DIAGNOSIS**: Kondisi medis yang ditentukan sebagai penyebab atau akibat dari gejala. Gejala dapat ditemukan melalui temuan fisik, laporan laboratorium atau radiologis, atau cara lain.
- **HYPOTHETICAL**: Indikasi bahwa suatu kondisi medis dinyatakan sebagai hipotesis.
- **LOW_CONFIDENCE**: Indikasi bahwa suatu kondisi medis dinyatakan memiliki ketidakpastian yang tinggi. Ini tidak terkait langsung dengan skor kepercayaan yang diberikan.
- **NEGATION**: Indikasi bahwa kondisi medis tidak ada.
- **PERTAINS_TO_FAMILY**: Indikasi bahwa suatu kondisi medis relevan dengan keluarga pasien, bukan pasien.
- **SIGN**: Kondisi medis yang dilaporkan oleh dokter.
- **SYMPTON**: Kondisi medis yang dilaporkan oleh pasien.

Kategori tes, perawatan, dan prosedur

TEST_TREATMENT_PROCEDURE kategori mendeteksi prosedur yang digunakan untuk menentukan kondisi medis.

Tipe

Untuk kategori **TEST_TREATMENT_PROCEDURE**, jenis berikut terdeteksi:

- **PROCEDURE_NAME**: Intervensi dilakukan pada pasien untuk mengobati kondisi medis atau untuk memberikan perawatan pasien.
- **TEST_NAME**: Prosedur yang dilakukan pada pasien untuk diagnosis, pengukuran, skrining, atau peringkat yang mungkin memiliki nilai yang dihasilkan. Ini termasuk prosedur, proses, evaluasi, atau penilaian apa pun untuk menentukan diagnosis, untuk mengesampingkan atau menemukan suatu kondisi, atau untuk mengukur atau menilai pasien.
- **TREATMENT_NAME**: Intervensi dilakukan untuk memerangi penyakit atau gangguan. Ini termasuk obat-obatan, seperti antivirus dan vaksinasi.

Atribut

Untuk kategori TEST_TREATMENT_PROCEDURE, atribut berikut terdeteksi:

- TEST_NAME : Tes diagnostik dilakukan.
- TEST_VALUE : Hasil numerik dari tes diagnostik.
- TEST_UNIT : Unit yang terkait dengan TEST_VALUE : hasil.
- PROCEDURE_NAME : Nama operasi atau prosedur medis yang dilakukan.
- TREATMENT_NAME : Nama pengobatan yang diberikan kepada pasien.

Sifat-sifat

- FUTUREIndikasi bahwa tes, perawatan, atau prosedur mengacu pada tindakan atau peristiwa yang akan terjadi setelah subjek catatan.
- HYPOTHETICALIndikasi bahwa tes, pengobatan, atau prosedur dinyatakan sebagai hipotesis
- NEGATIONIndikasi bahwa hasil atau tindakan negatif atau tidak dilakukan.
- PAST_HISTORYIndikasi bahwa tes, perawatan, atau prosedur berasal dari masa lalu pasien (sebelum pertemuan saat ini).

Detail CT SNOMED

Termasuk dalam respons JSON adalah detail CT SNOMED, yang mencakup informasi berikut:

- EDITION : Hanya edisi AS yang didukung.
- VERSIONDATE : Cap tanggal versi SNOMED CT yang digunakan.
- LANGUAGE : Analisis bahasa Inggris (US-EN) didukung.

Contoh masukan dan respons

Contoh berikut menunjukkan bagaimana InferSNOMEDCT operasi bekerja. Berikut ini adalah contoh teks masukan.

“BHEENT: Turbinat inferior boggy, Tidak ada lesi orofaring”

Untuk teks masukan contoh ini, InferSNOMEDCT operasi mengembalikan output berikut.

```

{
  "Entities": [
    {
      "Category": "ANATOMY",
      "BeginOffset": 0,
      "EndOffset": 5,
      "Text": "HEENT",
      "Traits": [],
      "SNOMEDCTConcepts": [
        {
          "Code": "69536005",
          "Score": 0.8183674812316895,
          "Description": "Head structure (body structure)"
        },
        {
          "Code": "429031000124106",
          "Score": 0.8062137961387634,
          "Description": "Review of systems, head, ear, eyes, nose and throat
(procedure)"
        },
        {
          "Code": "385383008",
          "Score": 0.7023276090621948,
          "Description": "Ear, nose and throat structure (body structure)"
        },
        {
          "Code": "64237003",
          "Score": 0.6886451840400696,
          "Description": "Structure of left half of head (body structure)"
        },
        {
          "Code": "113028003",
          "Score": 0.6595167517662048,
          "Description": "Ear, nose and throat examination (procedure)"
        }
      ],
      "Score": 0.9941003918647766,
      "Attributes": [],
      "Type": "SYSTEM_ORGAN_SITE",
      "Id": 0
    },
    {

```



```
"Category": "MEDICAL_CONDITION",
"BeginOffset": 8,
"EndOffset": 33,
"Text": "Boggy inferior turbinates",
"Traits": [
  {
    "Score": 0.916421115398407,
    "Name": "SIGN"
  }
],
"SNOMEDCTConcepts": [
  {
    "Code": "254477009",
    "Score": 0.3194539248943329,
    "Description": "Tumor of inferior turbinate (disorder)"
  },
  {
    "Code": "260762006",
    "Score": 0.2589553892612457,
    "Description": "Choroidal invasion status (attribute)"
  },
  {
    "Code": "2455009",
    "Score": 0.2561122477054596,
    "Description": "Revision of lumbosubarachnoid shunt (procedure)"
  },
  {
    "Code": "19883003",
    "Score": 0.25573015213012695,
    "Description": "Atrophy of nasal turbinates (disorder)"
  },
  {
    "Code": "256723009",
    "Score": 0.2551479935646057,
    "Description": "Inferior turbinate flap (substance)"
  }
],
"Score": 0.8120518326759338,
"Attributes": [
  {
    "Category": "ANATOMY",
    "RelationshipScore": 0.9952282905578613,
    "EndOffset": 5,
    "Text": "HEENT",
```

```

        "Traits": [],
        "SNOMEDCTConcepts": [
            {
                "Code": "69536005",
                "Score": 0.8183674812316895,
                "Description": "Head structure (body structure)"
            },
            {
                "Code": "429031000124106",
                "Score": 0.8062137961387634,
                "Description": "Review of systems, head, ear, eyes, nose
and throat (procedure)"
            },
            {
                "Code": "385383008",
                "Score": 0.7023276090621948,
                "Description": "Ear, nose and throat structure (body
structure)"
            },
            {
                "Code": "64237003",
                "Score": 0.6886451840400696,
                "Description": "Structure of left half of head (body
structure)"
            },
            {
                "Code": "113028003",
                "Score": 0.6595167517662048,
                "Description": "Ear, nose and throat examination
(procedure)"
            }
        ],
        "Score": 0.9941003918647766,
        "RelationshipType": "SYSTEM_ORGAN_SITE",
        "Type": "SYSTEM_ORGAN_SITE",
        "Id": 0,
        "BeginOffset": 0
    }
],
    "Type": "DX_NAME",
    "Id": 1
},
{
    "Category": "ANATOMY",

```

```

    "BeginOffset": 23,
    "EndOffset": 33,
    "Text": "turbinates",
    "Traits": [],
    "SNOMEDCTConcepts": [
      {
        "Code": "310607007",
        "Score": 0.38427865505218506,
        "Description": "Sarcoidosis of inferior turbinates (disorder)"
      },
      {
        "Code": "80153006",
        "Score": 0.35948991775512695,
        "Description": "Segmented neutrophil (cell)"
      },
      {
        "Code": "46607005",
        "Score": 0.34975120425224304,
        "Description": "Nasal turbinate structure (body structure)"
      },
      {
        "Code": "6553002",
        "Score": 0.3453119397163391,
        "Description": "Inferior nasal turbinate structure (body
structure)"
      },
      {
        "Code": "254477009",
        "Score": 0.34111809730529785,
        "Description": "Tumor of inferior turbinate (disorder)"
      }
    ],
    "Score": 0.6760638356208801,
    "Attributes": [],
    "Type": "SYSTEM_ORGAN_SITE",
    "Id": 3
  },
  {
    "Category": "ANATOMY",
    "BeginOffset": 39,
    "EndOffset": 52,
    "Text": "oropharyngeal",
    "Traits": [],
    "SNOMEDCTConcepts": [

```

```

    {
      "Code": "31389004",
      "Score": 0.8781343102455139,
      "Description": "Oropharyngeal structure (body structure)"
    },
    {
      "Code": "33431000119109",
      "Score": 0.865419328212738,
      "Description": "Lesion of oropharynx (disorder)"
    },
    {
      "Code": "263376008",
      "Score": 0.7922793626785278,
      "Description": "Entire oropharynx (body structure)"
    },
    {
      "Code": "716151000",
      "Score": 0.7752759456634521,
      "Description": "Structure of oropharynx and/or hypopharynx and/or
larynx (body structure)"
    },
    {
      "Code": "764786007",
      "Score": 0.7574880719184875,
      "Description": "Oropharyngeal (intended site)"
    }
  ],
  "Score": 0.33921703696250916,
  "Attributes": [],
  "Type": "SYSTEM_ORGAN_SITE",
  "Id": 5
},
{
  "Category": "MEDICAL_CONDITION",
  "BeginOffset": 39,
  "EndOffset": 59,
  "Text": "oropharyngeal lesion",
  "Traits": [
    {
      "Score": 0.925685465335846,
      "Name": "SIGN"
    }
  ]
},
"SNOMEDCTConcepts": [

```

```

    {
      "Code": "31389004",
      "Score": 0.8340228199958801,
      "Description": "Oropharyngeal structure (body structure)"
    },
    {
      "Code": "33431000119109",
      "Score": 0.830550491809845,
      "Description": "Lesion of oropharynx (disorder)"
    },
    {
      "Code": "764786007",
      "Score": 0.7099332213401794,
      "Description": "Oropharyngeal (intended site)"
    },
    {
      "Code": "418664002",
      "Score": 0.6987537741661072,
      "Description": "Oropharyngeal route (qualifier value)"
    },
    {
      "Code": "110162001",
      "Score": 0.6958084106445312,
      "Description": "Abrasion of oropharynx (disorder)"
    }
  ],
  "Score": 0.8390859961509705,
  "Attributes": [
    {
      "Category": "ANATOMY",
      "RelationshipScore": 0.9978047013282776,
      "EndOffset": 5,
      "Text": "HEENT",
      "Traits": [],
      "SNOMEDCTConcepts": [
        {
          "Code": "69536005",
          "Score": 0.8183674812316895,
          "Description": "Head structure (body structure)"
        },
        {
          "Code": "429031000124106",
          "Score": 0.8062137961387634,

```

```

        "Description": "Review of systems, head, ear, eyes, nose
and throat (procedure)"
      },
      {
        "Code": "385383008",
        "Score": 0.7023276090621948,
        "Description": "Ear, nose and throat structure (body
structure)"
      },
      {
        "Code": "64237003",
        "Score": 0.6886451840400696,
        "Description": "Structure of left half of head (body
structure)"
      },
      {
        "Code": "113028003",
        "Score": 0.6595167517662048,
        "Description": "Ear, nose and throat examination
(procedure)"
      }
    ],
    "Score": 0.9941003918647766,
    "RelationshipType": "SYSTEM_ORGAN_SITE",
    "Type": "SYSTEM_ORGAN_SITE",
    "Id": 0,
    "BeginOffset": 0
  }
],
  "Type": "DX_NAME",
  "Id": 4
}
],
  "SNOMEDCTDetails": {
    "Edition": "US",
    "VersionDate": "20200901",
    "Language": "en"
  },
  "Characters": {
    "OriginalTextCharacters": 59
  },
  "ModelVersion": "2.6.0.20220301"
}

```

Ontologi yang menghubungkan analisis batch

Gunakan Amazon Comprehend Medical untuk mendeteksi entitas dalam teks klinis yang disimpan dalam bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) dan untuk menautkan entitas tersebut ke ontologi standar. Anda dapat menggunakan ontologi yang menghubungkan analisis batch untuk menganalisis koleksi dokumen atau dokumen tunggal dengan hingga 20.000 karakter. Dengan menggunakan konsol atau ontologi yang menautkan operasi API batch, Anda dapat melakukan operasi untuk memulai, menghentikan, membuat daftar, dan menjelaskan pekerjaan analisis batch yang sedang berlangsung.

[Untuk informasi harga untuk analisis batch dan operasi Amazon Comprehend Medical lainnya, lihat Amazon Comprehend Medical Pricing.](#)

Melakukan analisis batch

Anda dapat menjalankan pekerjaan analisis batch menggunakan konsol Amazon Comprehend Medical atau operasi API batch Amazon Comprehend Medical.

Melakukan analisis batch menggunakan operasi API

Prasyarat

Saat Anda menggunakan Amazon Comprehend Medical API, buat kebijakan AWS Identity Access and Management (IAM) dan lampirkan ke peran IAM. Untuk mempelajari lebih lanjut tentang peran IAM dan kebijakan kepercayaan, lihat Kebijakan [dan Izin IAM](#).

1. Unggah data Anda ke dalam bucket S3.
2. Untuk memulai pekerjaan analisis baru, gunakan StartICD10CM, InferenceJobStartsNomeDCT InferenceJob, atau operasi. StartRxNormInferenceJob Berikan nama bucket Amazon S3 yang berisi file input dan nama bucket Amazon S3 tempat Anda ingin mengirim file output.
3. Pantau kemajuan pekerjaan dengan menggunakan DescribeICD10CM, DescribesNoMEDCT InferenceJob, atau operasi. InferenceJob DescribeRxNormInferenceJob Selain itu, Anda dapat menggunakan ListICD10cm InferenceJobs, ListsNomeDCT InferenceJobs, dan ListRxNormInferenceJobs untuk melihat status semua ontologi yang menghubungkan pekerjaan analisis batch.

4. Jika Anda perlu menghentikan pekerjaan yang sedang berlangsung, gunakan `stopicd10cm`, `InferenceJobstopsnomedCT InferenceJob`, atau untuk menghentikan analisis. `StopRxNormInferenceJob`
5. Untuk melihat hasil pekerjaan analisis Anda, lihat bucket S3 keluaran yang Anda konfigurasi saat memulai pekerjaan.

Melakukan analisis batch menggunakan konsol

1. Unggah data Anda ke dalam bucket S3.
2. Untuk memulai pekerjaan analisis baru, pilih jenis analisis yang akan Anda lakukan. Kemudian, berikan nama bucket S3 yang berisi file input dan nama bucket S3 tempat Anda ingin mengirim file output.
3. Pantau status pekerjaan Anda saat sedang berlangsung. Dari konsol, Anda dapat melihat semua operasi analisis batch dan statusnya, termasuk kapan analisis dimulai dan berakhir.
4. Untuk melihat hasil pekerjaan analisis Anda, lihat bucket keluaran S3 yang Anda konfigurasi saat memulai pekerjaan.

Kebijakan IAM untuk operasi batch

Peran IAM yang memanggil operasi API batch Amazon Comprehend Medical harus memiliki kebijakan yang memberikan akses ke bucket S3 yang berisi file input dan output. Peran IAM juga harus diberi hubungan kepercayaan sehingga layanan Amazon Comprehend Medical dapat mengambil peran tersebut. Untuk mempelajari lebih lanjut tentang peran IAM dan kebijakan kepercayaan, lihat Peran [IAM](#).

Peran harus memiliki kebijakan berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::input-bucket/*"
      ]
    }
  ]
}
```



```

    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:ListBucket"
    ],
    "Resource": [
      "arn:aws:s3:::input-bucket",
      "arn:aws:s3:::output-bucket",
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:PutObject"
    ],
    "Resource": [
      "arn:aws:s3:::output-bucket/*"
    ],
    "Effect": "Allow"
  }
]
}

```

Peran tersebut harus memiliki hubungan kepercayaan berikut. Disarankan agar Anda menggunakan kunci `aws:SourceAccount` dan `aws:SourceArn` kondisi untuk mencegah masalah keamanan wakil yang membingungkan. Untuk mempelajari lebih lanjut tentang masalah wakil yang bingung dan cara melindungi AWS akun Anda, lihat [Masalah wakil yang bingung](#) dalam dokumentasi IAM.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "comprehendmedical.amazonaws.com"
        ]
      }
    }
  ]
}

```

```

    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "account_id"
      },
      "ArnLike": {
        "aws:SourceArn": [
          "arn:aws:s3:::input-bucket/*",
          "arn:aws:s3:::output-bucket"
        ]
      }
    }
  }
]
}

```

File keluaran analisis Batch

Amazon Comprehend Medical membuat satu file output untuk setiap file input dalam batch. File memiliki ekstensi `.out`. Amazon Comprehend Medical pertama-tama membuat direktori di bucket output S3 menggunakan `AwsAccountIdJobType- JobId` sebagai namanya, dan kemudian menulis semua file output untuk batch ke direktori ini. Amazon Comprehend Medical membuat direktori baru ini sehingga output dari satu pekerjaan tidak menimpa output dari pekerjaan lain.

Operasi batch menghasilkan output yang sama dengan operasi sinkron.

Setiap operasi batch menghasilkan tiga file manifes berikut yang berisi informasi tentang pekerjaan:

- **Manifest**— Meringkas pekerjaan. Memberikan informasi tentang parameter yang digunakan untuk pekerjaan, ukuran total pekerjaan, dan jumlah file yang diproses.
- **Success**— Memberikan informasi tentang file yang berhasil diproses. Termasuk nama file input dan output dan ukuran file input.
- **Unprocessed**— Daftar file yang tidak diproses oleh pekerjaan batch. Biasanya, file tidak diproses karena ditambahkan ke direktori input setelah pekerjaan batch dimulai.

Amazon Comprehend Medical menulis file-file ini ke direktori output yang Anda tentukan untuk pekerjaan batch. Bagian berikut menunjukkan struktur file manifes.

File manifes Batch

Berikut ini adalah struktur JSON dari file manifes batch.

```
{
  "Summary" : {
    "Status" : "COMPLETED | FAILED | PARTIAL_SUCCESS | STOPPED",
    "JobType" : "ICD10CMInferenceJob | RxNormInferenceJob | SNOMEDCTInferenceJob",
    "InputDataConfiguration" : {
      "Bucket" : "input bucket",
      "Path" : "path to files/account ID-job type-job ID"
    },
    "OutputDataConfiguration" : {
      "Bucket" : "output bucket",
      "Path" : "path to files"
    },
    "InputFileCount" : number of files in input bucket,
    "TotalMeteredCharacters" : total characters processed from all files,
    "UnprocessedFilesCount" : number of files not processed,
    "SuccessFilesCount" : total number of files processed,
    "TotalDurationSeconds" : time required for processing,
    "SuccessfulFilesListLocation" : "path to file",
    "UnprocessedFilesListLocation" : "path to file"
  }
}
```

File manifes sukses

Berikut ini adalah struktur JSON dari file yang berisi informasi tentang file yang berhasil diproses.

```
{
  "Files": [{
    "Input": "input path/input file name",
    "Output": "output path/output file name",
    "InputSize": size in bytes of input file
  },
  {
    "Input": "input path/input file name",
    "Output": "output path/output file name",
    "InputSize": size in bytes of input file
  }
]
```

File manifes yang belum diproses

Berikut ini adalah struktur JSON dari file manifes yang berisi informasi tentang file yang belum diproses.

```
{
  "Files": [
    "input path/input file name",
    "input path/input file name"
  ]
}
```

Keamanan di Amazon Comprehend Medical

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. 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 Comprehend Medical, [lihat AWS Services in Scope by Compliance Program AWS Services in Scope by Compliance](#) .
- 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 Comprehend Medical. Topik berikut menunjukkan cara mengkonfigurasi Comprehend Medical untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan layanan AWS lain yang membantu Anda memantau dan mengamankan sumber daya Comprehend Medical Anda.

Topik

- [Perlindungan data di Amazon Comprehend Medical](#)
- [Manajemen identitas dan akses di Amazon Comprehend Medical](#)
- [Logging panggilan API Amazon Comprehend Medical menggunakan AWS CloudTrail](#)
- [Validasi kepatuhan untuk Amazon Comprehend Medical](#)
- [Ketahanan di Amazon Comprehend Medical](#)
- [Keamanan infrastruktur di Amazon Comprehend Medical](#)

Perlindungan data di Amazon Comprehend Medical

[Model tanggung jawab AWS bersama model](#) berlaku untuk perlindungan data di Amazon Comprehend Medical. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda harus bertanggung jawab untuk memelihara kendali terhadap konten yang di-hosting pada infrastruktur ini. Konten ini meliputi konfigurasi keamanan dan tugas-tugas pengelolaan untuk berbagai layanan 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 kredensial 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 sumber daya. AWS Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-2 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi yang lebih lengkap tentang titik akhir FIPS yang tersedia di [Standar Pemrosesan Informasi Federal \(FIPS\) 140-2](#).

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 Comprehend Medical atau Layanan AWS lainnya menggunakan konsol, API, atau SDK. AWS CLI AWS 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.

Manajemen identitas dan akses di Amazon Comprehend Medical

Akses ke Comprehend Medical memerlukan kredensial yang dapat digunakan AWS untuk mengautentikasi permintaan Anda. Kredensial tersebut harus memiliki izin untuk mengakses Comprehend Medical action. [AWS Identity and Access Management \(IAM\)](#) dapat membantu mengamankan sumber daya Anda dengan mengendalikan siapa yang dapat mengaksesnya. Bagian berikut memberikan detail tentang cara Anda menggunakan IAM dengan Comprehend Medical.

- [Autentikasi](#)
- [Kontrol Akses](#)

Autentikasi

Anda harus memberikan izin kepada pengguna untuk berinteraksi dengan Amazon Comprehend Medical. Untuk pengguna yang membutuhkan penggunaan akses penuh `ComprehendMedicalFullAccess`.

Untuk menyediakan akses, tambahkan izin ke pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat set izin. Ikuti petunjuk di [Buat set izin](#) di Panduan AWS IAM Identity Center Pengguna.

- Pengguna yang dikelola dalam IAM melalui penyedia identitas:

Membuat peran untuk federasi identitas. Ikuti petunjuk dalam [Membuat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) di Panduan Pengguna IAM.

- Pengguna IAM:

- Buat peran yang dapat diasumsikan pengguna Anda. Ikuti petunjuk dalam [Membuat peran untuk pengguna IAM](#) di Panduan Pengguna IAM.
- (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam [Menambahkan izin ke pengguna \(konsol\)](#) di Panduan Pengguna IAM.

Untuk menggunakan operasi asinkron Amazon Comprehend Medical, Anda juga memerlukan peran layanan.

Peran layanan adalah [IAM role](#) yang diambil oleh layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi lebih lanjut, lihat [Membuat peranLayanan AWS untuk mendelegasikan izin ke Panduan](#).

Untuk mempelajari selengkapnya tentang menentukan Amazon Comprehend Medical sebagai layanan di kepala sekolah, lihat [Izin berbasis peran diperlukan untuk operasi batch](#).

Kontrol Akses

Anda harus memiliki kredenitas yang sah untuk mengautentikasi permintaan Anda. Kredensi harus memiliki izin untuk memanggil tindakan Amazon Comprehend Medical.

Bagian berikut menjelaskan cara mengelola izin untuk Amazon Comprehend Medical. Kami menyarankan agar Anda membaca gambaran umum terlebih dahulu.

- [Gambaran umum pengelolaan izin akses untuk sumber daya Amazon Comprehend Medical](#)
- [Menggunakan kebijakan berbasis identitas \(kebijakan IAM\) untuk Amazon Comprehend Medical](#)

Topik

- [Gambaran umum pengelolaan izin akses untuk sumber daya Amazon Comprehend Medical](#)
- [Menggunakan kebijakan berbasis identitas \(kebijakan IAM\) untuk Amazon Comprehend Medical](#)
- [Izin Amazon Comprehend Medical: tindakan, sumber daya, dan referensi kondisi](#)

Gambaran umum pengelolaan izin akses untuk sumber daya Amazon Comprehend Medical

Kebijakan izin mengatur akses ke suatu tindakan. Administrator akun melampirkan kebijakan izin ke identitas IAM untuk mengelola akses ke tindakan. Identitas IAM termasuk pengguna, grup, dan peran.

Note

Seorang administrator akun (atau pengguna administrator) adalah pengguna dengan hak istimewa administrator. Untuk informasi lebih lanjut, lihat [Praktik Terbaik IAM](#) di Panduan Pengguna IAM.

Saat Anda memberikan izin, Anda memutuskan siapa dan tindakan apa yang mendapatkan izin.

Topik

- [Mengelola akses ke tindakan](#)
- [Menentukan elemen kebijakan: tindakan, efek, dan prinsip](#)
- [Menentukan syarat dalam kebijakan](#)

Mengelola akses ke tindakan

Kebijakan izin menjelaskan siapa yang memiliki akses ke objek. Bagian berikut menjelaskan pilihan kebijakan izin.

Note

Bagian ini menjelaskan IAM dalam konteks Amazon Comprehend Medical. Bagian ini tidak memberikan informasi detail tentang layanan IAM. Untuk informasi lebih lanjut tentang IAM, lihat [Apa itu IAM?](#) dalam Panduan Pengguna IAM. Untuk informasi tentang sintaksis dan penjelasan kebijakan [IAM](#), lihat [Referensi Kebijakan IAM](#).

Kebijakan yang terlampir pada identitas IAM adalah kebijakan berbasis identitas. Kebijakan yang terlampir pada sumber daya adalah kebijakan berbasis sumber daya. Amazon Comprehend Medical hanya mendukung kebijakan berbasis identitas.

Kebijakan berbasis identitas (Kebijakan IAM)

Anda dapat melampirkan kebijakan ke identitas IAM. Berikut ini adalah dua contoh.

- Lampirkan kebijakan izin ke pengguna atau grup di akun Anda. Untuk memberikan pengguna atau grup pengguna untuk memanggil tindakan Amazon Comprehend Medical, lampirkan kebijakan izin ke pengguna. Lampirkan kebijakan ke grup yang berisi pengguna.

- Lampirkan kebijakan izin ke peran untuk memberikan izin lintas akun. Untuk memberikan izin lintas-akun, lampirkan kebijakan berbasis identitas ke IAM role. Misalnya, administrator pada akun A dapat membuat peran untuk memberikan izin lintas akun ke akun lain. Dalam contoh ini, sebut saja Akun B, yang juga bisa menjadi layanan AWS.
 1. Administrator akun A membuat IAM role dan melampirkan kebijakan pada peran yang memberikan izin ke sumber daya di Akun A.
 2. Akun Administrator melampirkan kebijakan kepercayaan pada peran tersebut. Kebijakan ini mengidentifikasi Akun B sebagai penanggung jawab yang dapat menjalankan peran tersebut.
 3. Administrator Akun B kemudian dapat mendelegasikan izin untuk mengasumsikan peran pada pengguna dalam akun B. Dengan melakukan ini, pengguna dalam akun B dapat membuat atau mengakses sumber daya di Akun A. Jika Anda ingin memberikan izin layanan AWS.

Untuk informasi selengkapnya tentang menggunakan IAM untuk mendelegasikan izin, lihat [Manajemen Akses](#) dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang menggunakan kebijakan berbasis identitas dengan Amazon Comprehend Medical, lihat [Menggunakan kebijakan berbasis identitas \(kebijakan IAM\) untuk Amazon Comprehend Medical](#). Untuk informasi lebih lanjut tentang pengguna, grup, peran, dan izin, lihat [Identitas \(Pengguna, Grup, dan Peran\)](#) dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Layanan lain, seperti AWS Lambda, mendukung kebijakan izin berbasis sumber daya. Misalnya, Anda dapat melampirkan kebijakan ke bucket S3 untuk mengelola izin akses ke bucket tersebut. Amazon Comprehend Medical tidak mendukung kebijakan berbasis sumber daya.

Menentukan elemen kebijakan: tindakan, efek, dan prinsip

Amazon Comprehend Medical menentukan serangkaian operasi API. Untuk memberikan izin bagi operasi API ini, Amazon Comprehend Medical menentukan serangkaian tindakan yang dapat Anda tentukan dalam kebijakan.

Empat item di sini adalah elemen-elemen paling dasar.

- Sumber Daya — Dalam kebijakan, gunakan Amazon Resource Name (ARN) untuk mengidentifikasi sumber daya yang menjadi tempat kebijakan itu berlaku. Untuk Amazon Comprehend Medical, sumber daya selalu "*" .

- **Tindakan** — Gunakan kata kunci tindakan untuk mengidentifikasi operasi yang ingin Anda izinkan atau tolak. Misalnya, tergantung pada efek yang ditentukan, mengizinkan `comprehendmedical:DetectEntities` atau menolak izin pengguna untuk melakukan `DetectEntities` operasi Amazon Comprehend Medical.
- **Efek** — Tentukan efek tindakan yang terjadi ketika pengguna meminta tindakan tertentu— baik mengizinkan maupun menolak. Jika Anda tidak secara eksplisit memberikan akses untuk (mengizinkan) sumber daya, akses akan ditolak secara implisit. Anda juga dapat secara eksplisit menolak akses ke sumber daya. Anda dapat melakukan ini untuk memastikan bahwa pengguna tidak dapat mengakses sumber daya, meskipun kebijakan yang berbeda memberikan akses.
- **Prinsipal** — Dalam kebijakan berbasis identitas, pengguna yang kebijakannya terlampir adalah prinsipal implisit.

Untuk mempelajari selengkapnya tentang sintaks dan penjelasan kebijakan IAM, lihat [AWS IAM Policy Reference](#) dalam Panduan Pengguna IAM.

Untuk tabel yang menunjukkan semua tindakan API Amazon Comprehend Medical, lihat [Izin Amazon Comprehend Medical: tindakan, sumber daya, dan referensi kondisi](#).

Menentukan syarat dalam kebijakan

Saat Anda memberikan izin, Anda menggunakan bahasa kebijakan IAM untuk menentukan syarat di mana kebijakan akan berlaku. Misalnya, Anda mungkin ingin kebijakan diterapkan hanya setelah tanggal tertentu. Untuk informasi selengkapnya tentang menentukan syarat dalam bahasa kebijakan, lihat [Syarat](#) dalam Panduan Pengguna IAM.

AWS menyediakan serangkaian kunci syarat yang telah ditetapkan untuk semua layanan AWS yang mendukung IAM untuk kontrol akses. Misalnya, Anda dapat menggunakan kunci `aws:user:uid` syarat untuk mewajibkan ID AWS tertentu saat meminta tindakan. Untuk informasi selengkapnya dan daftar lengkap kunci AWS AWS, lihat [Kunci yang Tersedia untuk Kondisi](#) dalam Panduan.

Amazon Comprehend Medical tidak menyediakan kunci kondisi tambahan apa pun.

Menggunakan kebijakan berbasis identitas (kebijakan IAM) untuk Amazon Comprehend Medical

Topik ini menunjukkan contoh kebijakan berbasis identitas. Contoh menunjukkan cara administrator akun dapat melampirkan kebijakan izin ke identitas IAM. Hal ini memungkinkan pengguna, grup, dan peran untuk melakukan tindakan Amazon Comprehend Medical.

⚠ Important

Untuk memahami izin, kami sarankan [Gambaran umum pengelolaan izin akses untuk sumber daya Amazon Comprehend Medical](#).

Kebijakan contoh ini diperlukan untuk menggunakan tindakan analisis dokumen Amazon Comprehend Medical.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowDetectActions",
    "Effect": "Allow",
    "Action": [
      "comprehendmedical:DetectEntitiesV2",
      "comprehendmedical:DetectEntities",
      "comprehendmedical:DetectPHI",

      "comprehendmedical:StartEntitiesDetectionV2Job",
      "comprehendmedical:ListEntitiesDetectionV2Jobs",
      "comprehendmedical:DescribeEntitiesDetectionV2Job",
      "comprehendmedical:StopEntitiesDetectionV2Job",

      "comprehendmedical:StartPHIDetectionJob",
      "comprehendmedical:ListPHIDetectionJobs",
      "comprehendmedical:DescribePHIDetectionJob",
      "comprehendmedical:StopPHIDetectionJob",

      "comprehendmedical:StartRxNormInferenceJob",
      "comprehendmedical:ListRxNormInferenceJobs",
      "comprehendmedical:DescribeRxNormInferenceJob",
      "comprehendmedical:StopRxNormInferenceJob",

      "comprehendmedical:StartICD10CMInferenceJob",
      "comprehendmedical:ListICD10CMInferenceJobs",
      "comprehendmedical:DescribeICD10CMInferenceJob",
      "comprehendmedical:StopICD10CMInferenceJob",

      "comprehendmedical:StartSNOMEDCTInferenceJob",
      "comprehendmedical:ListSNOMEDCTInferenceJobs",
      "comprehendmedical:DescribeSNOMEDCTInferenceJob",
```

```

        "comprehendmedical:StopSNOMEDCTInferenceJob",

        "comprehendmedical:InferRxNorm",
        "comprehendmedical:InferICD10CM",
        "comprehendmedical:InferSNOMEDCT",

    ],
    "Resource": "*"
  }
]
}

```

Kebijakan ini memiliki satu pernyataan yang memberikan izin untuk menggunakan `DetectEntities` dan `DetectPHI` tindakan.

Kebijakan tidak menyebutkan `Principal` elemen karena Anda tidak menyebutkan prinsipal yang mendapatkan izin dalam kebijakan berbasis identitas. Saat Anda melampirkan kebijakan kepada pengguna, pengguna adalah penanggung jawab implisit. Saat Anda melampirkan kebijakan pada IAM role, prinsipal yang diidentifikasi dalam kebijakan kepercayaan peran mendapatkan izin.

Untuk melihat semua tindakan API Amazon Comprehend Medical dan sumber daya yang diterapkan, lihat [Izin Amazon Comprehend Medical: tindakan, sumber daya, dan referensi kondisi](#).

Izin diperlukan untuk menggunakan konsol Amazon Comprehend Medical

Tabel referensi perizinan mencantumkan operasi Amazon Comprehend Medical API dan menunjukkan izin yang diperlukan untuk setiap operasi. Untuk informasi selengkapnya, tentang izin Amazon Comprehend Medical API, lihat [Izin Amazon Comprehend Medical: tindakan, sumber daya, dan referensi kondisi](#).

Untuk menggunakan konsol Amazon Comprehend Medical, berikan izin untuk tindakan yang ditunjukkan dalam kebijakan berikut.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateRole",

```

```

        "iam:CreatePolicy",
        "iam:AttachRolePolicy"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": "iam:PassRole",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "comprehendmedical.amazonaws.com"
      }
    }
  }
]
}

```

Konsol Amazon Comprehend Medical memerlukan izin ini karena alasan berikut:

- iamizin untuk mencantumkan peran IAM yang tersedia untuk akun Anda.
- s3izin untuk mengakses bucket Amazon S3 dan objek yang berisi data.

Saat membuat tugas batch asinkron menggunakan konsol, Anda juga dapat membuat IAM role untuk tugas Anda. Untuk membuat peran IAM menggunakan konsol, pengguna harus diberikan izin tambahan yang ditampilkan di sini untuk membuat peran dan kebijakan IAM, dan untuk melampirkan kebijakan pada peran.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "iam:CreateRole",
        "iam:CreatePolicy",
        "iam:AttachRolePolicy"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}

```

```
]
}
```

Konsol Amazon Comprehend Medical memerlukan izin ini untuk membuat peran dan kebijakan serta untuk melampirkan peran dan kebijakan. `iam:PassRoleTindakan` ini memungkinkan konsol untuk meneruskan peran ke Amazon Comprehend Medical.

Kebijakan terkelola AWS (ditentukan sebelumnya) untuk Amazon Comprehend Medical

AWS menangani banyak kasus penggunaan umum dengan menyediakan kebijakan IAM mandiri yang dibuat dan dikelola oleh AWS. Kebijakan terkelola AWS ini memberikan izin yang diperlukan untuk kasus penggunaan umum sehingga Anda dapat menghindari untuk menyelidiki izin apa yang diperlukan. Untuk informasi selengkapnya, lihat [Kebijakan yang Dikelola AWS](#) dalam Panduan Pengguna IAM.

Kebijakan terkelola AWS berikut, yang dapat Anda lampirkan ke pengguna di akun Anda, khusus untuk Amazon Comprehend Medical.

- `ComprehendMedicalFullAccess`— Memberi akses penuh ke sumber daya Amazon Comprehend Medical. Termasuk izin untuk daftar dan mendapatkan peran IAM.

Anda harus menerapkan kebijakan tambahan berikut ini kepada pengguna mana pun yang menggunakan Amazon Comprehend Medical:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "comprehendmedical.amazonaws.com"
        }
      }
    }
  ]
}
```

```
]
}
```

Anda dapat meninjau kebijakan izin terkelola dengan masuk ke konsol IAM dan mencari kebijakan tertentu di sana.

Kebijakan ini berlaku ketika Anda menggunakan SDK AWS, atau AWS CLI.

Anda juga dapat membuat kebijakan IAM Anda sendiri untuk memberikan izin untuk tindakan dan sumber daya Amazon Comprehend Medical. Anda dapat melampirkan kebijakan kustom ini ke pengguna IAM atau grup yang memerlukannya.

Izin berbasis peran diperlukan untuk operasi batch

Untuk menggunakan operasi asinkron Amazon Comprehend Medical, berikan akses Amazon Comprehend Medical ke bucket Amazon S3 yang berisi koleksi dokumen Anda. Lakukan ini dengan membuat peran akses data di akun Anda untuk mempercayai prinsipal layanan Amazon Comprehend Medical. Untuk informasi lebih lanjut tentang cara membuat peran, lihat [Membuat Peran untuk Mendelegasikan Izin ke Layanan](#) AWS dalam Panduan Pengguna.

Berikut ini adalah kebijakan kepercayaan peran.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "comprehendmedical.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Setelah Anda membuat peran, buat kebijakan akses untuk itu. Kebijakan tersebut harus memberikan Amazon S3GetObject dan ListBucket izin ke bucket Amazon S3 yang berisi data masukan Anda. Ini juga memberikan izin untuk Amazon S3PutObject ke bucket data keluaran Amazon S3 Anda.

Kebijakan akses contoh berikut berisi izin tersebut.


```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::input bucket/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::input bucket"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:PutObject"
      ],
      "Resource": [
        "arn:aws:s3:::output bucket/*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

Contoh kebijakan yang dikelola pelanggan

Dalam bagian ini, Anda dapat menemukan contoh kebijakan pengguna yang memberikan izin untuk berbagai tindakan Amazon Comprehend Medical. Kebijakan ini berlaku ketika Anda menggunakan SDK AWS, atau AWS CLI. Saat menggunakan konsol, Anda harus memberikan izin ke semua API Amazon Comprehend Medical. Hal ini dibahas dalam [Izin diperlukan untuk menggunakan konsol Amazon Comprehend Medical](#).

Note

Semua contoh menggunakan wilayah us-timur-2 dan berisi ID akun fiktif.

Contoh**Contoh 1: Mengizinkan semua tindakan Amazon Comprehend Medical**

Setelah mendaftar AWS, Anda membuat administrator untuk mengelola akun, termasuk membuat pengguna dan mengelola izin mereka.

Anda dapat memilih untuk membuat pengguna yang memiliki izin untuk semua tindakan Amazon Comprehend. Pikirkan pengguna ini sebagai administrator khusus layanan untuk bekerja dengan Amazon Comprehend. Anda dapat melampirkan kebijakan izin berikut ke pengguna ini.

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

Contoh 2: Hanya Mengizinkan DetectEntities Tindakan

Kebijakan izin berikut memberikan izin kepada pengguna untuk mendeteksi entitas di Amazon Comprehend Medical, tetapi tidak untuk mendeteksi operasi PHI.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowDetectEntityActions",
    "Effect": "Allow",
    "Action": [
      "comprehendmedical:DetectEntities"
    ]
  }
]
```

```
    ],  
    "Resource": "*" ]  
  }  
]
```

Izin Amazon Comprehend Medical: tindakan, sumber daya, dan referensi kondisi

Gunakan tabel berikut sebagai referensi ketika menyiapkan [Kontrol Akses](#) dan menulis kebijakan izin yang dapat Anda lampirkan ke pengguna. daftar tersebut mencakup masing-masing Amazon Comprehend Medical API, tindakan terkait yang dapat Anda berikan izin untuk melakukan tindakan tersebut, dan sumber daya AWS di mana Anda dapat memberikan izin. Anda menentukan tindakan di kolom `Action` kebijakan, dan Anda menentukan nilai sumber daya di kolom `Resource` kebijakan.

Untuk menyatakan syarat, Anda dapat menggunakan kunci syarat AWS di kebijakan Amazon Comprehend Medical Anda. Untuk daftar lengkap kunci, lihat [Kunci yang Tersedia](#) di Panduan Pengguna IAM.

Note

Untuk menentukan tindakan, gunakan `comprehendmedical:` awalan diikuti dengan nama operasi API, misalnya, `comprehendmedical:DetectEntities`.

Logging panggilan API Amazon Comprehend Medical menggunakan AWS CloudTrail

Amazon Comprehend Medical terintegrasi dengan AWS CloudTrail. CloudTrail adalah layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan dari dalam Amazon Comprehend Medical. CloudTrail menangkap semua panggilan API untuk Amazon Comprehend Medical sebagai kejadian. Panggilan yang ditangkap mencakup panggilan dari konsol Amazon Comprehend Medical dan panggilan kode ke operasi Amazon Comprehend Medical. Jika membuat jejak, Anda dapat mengaktifkan pengiriman peristiwa CloudTrail berkelanjutan ke bucket Amazon S3, termasuk peristiwa untuk Amazon Comprehend Medical. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru dalam konsol CloudTrail di Riwayat peristiwa.

Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan beberapa hal seperti:

- Permintaan yang dibuat untuk Amazon Comprehend Medical
- Alamat IP di mana permintaan itu dibuat
- Siapa yang membuat permintaan
- Ketika permintaan dibuat
- Detail lainnya

Untuk mempelajari selengkapnya tentang CloudTrail, lihat [Panduan Pengguna AWS CloudTrail](#).

Informasi Amazon Comprehend Medical di CloudTrail

CloudTrail diaktifkan pada akun AWS Anda saat Anda membuat akun tersebut. Saat aktivitas terjadi di Amazon Comprehend Medical, aktivitas tersebut dicatat di CloudTrail bersama lainnya AWS peristiwa layanan di Riwayat peristiwa. Anda dapat melihat, mencari, dan mengunduh peristiwa terbaru di akun AWS Anda. Untuk informasi selengkapnya, lihat [Melihat Peristiwa dengan Riwayat Peristiwa CloudTrail](#).

Untuk catatan berkelanjutan tentang peristiwa di AWS akun, termasuk peristiwa untuk Amazon Comprehend Medical, membuat jejak. Sebuah Jejak mengaktifkan CloudTrail untuk mengirim berkas log ke bucket Amazon S3. Secara default, ketika Anda membuat jejak di konsol tersebut, jejak tersebut diterapkan ke semua Wilayah AWS. Log acara jejak dari semua Wilayah di partisi AWS dan mengirimkan berkas log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi layanan AWS lainnya untuk menganalisis lebih lanjut dan bertindak berdasarkan data peristiwa yang dikumpulkan di log CloudTrail. Untuk informasi selengkapnya, lihat yang berikut:

- [Ikhtisar untuk Membuat Jejak](#)
- [Layanan yang Didukung dan Integrasi CloudTrail](#)
- [Mengonfigurasi Notifikasi Amazon SNS untuk CloudTrail](#)
- [Menerima Berkas Log CloudTrail dari Berbagai Wilayah](#) dan [Menerima Berkas Log CloudTrail dari Berbagai Akun](#)

Semua tindakan Amazon Comprehend Medical dicatat oleh CloudTrail dan didokumentasikan di [Referensi Amazon Comprehend Medical](#). Misalnya, panggilan untuk tindakan

DetectEntitiesV2, DetectPHI dan ListEntitiesDetectionV2Jobs menghasilkan entri di berkas log CloudTrail.

Setiap entri peristiwa atau log berisi informasi tentang orang yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut:

- Bahwa permintaan dibuat dengan kredensial pengguna root atau pengguna AWS Identity and Access Management (IAM).
- Bahwa permintaan tersebut dibuat dengan kredensial keamanan sementara untuk peran atau pengguna gabungan.
- Apakah permintaan dibuat oleh layanan AWS lain.

Untuk informasi lebih lanjut, lihat [Elemen userIdentity CloudTrail](#).

Memahami entri file log Amazon Comprehend Medical

Jejak adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai berkas log ke bucket Amazon S3 yang telah Anda tentukan. File log CloudTrail berisi satu atau beberapa entri log. Peristiwa menunjukkan satu permintaan dari sumber mana pun. Peristiwa mencakup informasi tentang tindakan yang diminta, seperti parameter tanggal dan waktu atau permintaan. Berkas log CloudTrail bukanlah pelacakan tumpukan terurut dari panggilan API publik, sehingga tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri log CloudTrail yang menunjukkan tindakan DetectEntitiesV2.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::123456789012:user/Mateo_Jackson",
    "accountId": "123456789012",
    "accessKeyId": "ASIAXHKUF0DNN8EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/Mateo_Jackson",
        "accountId": "123456789012",
```

```
        "userName": "Mateo_Jackson"
    },
    "webIdFederationData": {},
    "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-09-27T20:07:27Z"
    }
}
},
"eventTime": "2019-09-27T20:10:26Z",
"eventSource": "comprehendmedical.amazonaws.com",
"eventName": "DetectEntitiesV2",
"awsRegion": "us-east-1",
"sourceIPAddress": "702.21.198.166",
"userAgent": "aws-internal/3 aws-sdk-java/1.11.590
Linux/4.9.184-0.1.ac.235.83.329.metal1.x86_64 OpenJDK_64-Bit_Server_VM/25.212-b03
java/1.8.0_212 vendor/Oracle_Corporation",
"requestParameters": null,
"responseElements": null,
"requestID": "8d85f2ec-EXAMPLE",
"eventID": "ae9be9b1-EXAMPLE",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}
```

Validasi kepatuhan untuk Amazon Comprehend Medical

Auditor pihak ketiga menilai keamanan dan kepatuhan Amazon Comprehend Medical sebagai bagian dari beberapa program kepatuhan. AWS Ini termasuk PCI, FedRAMP, HIPAA, dan lainnya. 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 Comprehend Medical ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- [Panduan Quick Start Keamanan dan Kepatuhan](#) – Panduan deployment ini membahas pertimbangan arsitektur dan memberikan langkah untuk menerapkan lingkungan dasar yang berfokus pada keamanan dan kepatuhan di AWS.

- [Arsitektur untuk Whitepaper Keamanan dan Kepatuhan HIPAA — Whitepaper](#) ini menjelaskan bagaimana perusahaan dapat menggunakan untuk membuat aplikasi yang sesuai dengan HIPAA. AWS
- [AWS Sumber Daya Kepatuhan](#) — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- [AWS Config](#) AWS Layanan ini 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.

Untuk daftar AWS layanan dalam lingkup program kepatuhan tertentu, lihat [AWS Services in Scope by Compliance Program](#). Untuk informasi umum, lihat [Program Kepatuhan AWS](#).

Ketahanan di Amazon Comprehend Medical

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 dan mengoperasikan aplikasi dan basis data yang melakukan secara otomatis pinda saat gagal/failover di antara zona-zona tanpa terputus. Zona Ketersediaan lebih sangat tersedia, lebih toleran kesalahan, dan lebih dapat diskalakan daripada infrastruktur pusat data tunggal atau multi tradisional.

Untuk informasi selengkapnya tentang AWS Wilayah dan Availability Zone, lihat [Infrastruktur AWS Global](#).

Keamanan infrastruktur di Amazon Comprehend Medical

Sebagai layanan terkelola, Amazon Comprehend Medical dilindungi oleh prosedur keamanan jaringan global yang dijelaskan AWS dalam whitepaper Amazon Web [Services](#): Overview of Security Processes.

Untuk mengakses Comprehend Medical melalui jaringan, Anda menggunakan panggilan API yang dipublikasikan. AWS Klien harus mendukung Keamanan Lapisan Pengangkutan (TLS) 1.0 atau versi yang lebih baru. Kami merekomendasikan TLS 1.2 atau versi yang lebih baru. Selain itu, klien harus

mendukung cipher suites dengan perfect forward secrecy (PFS) seperti Ephemeral Diffie-Hellman (DHE) atau Elliptic Curve Diffie-Hellman Ephemeral (ECDHE). Sebagian besar sistem modern, misalnya Java 7 dan versi yang lebih baru, mendukung mode ini.

Selain itu, permintaan harus ditandatangani dengan menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan prinsipal AWS Identity and Access Management (IAM). Atau Anda bisa menggunakan [AWS Security Token Service](#) (AWS STS) untuk membuat kredensial keamanan sementara guna menandatangani permintaan.

Pedoman dan kuota

Ingatlah informasi berikut saat menggunakan Amazon Comprehend Medical.

Hanya dokumen dalam bahasa Inggris (EN) yang didukung.

Pemberitahuan penting

Amazon Comprehend Medical bukan pengganti saran medis profesional, diagnosis, atau perawatan. Amazon Comprehend Medical memberikan skor kepercayaan yang menunjukkan tingkat kepercayaan pada keakuratan entitas yang terdeteksi. Identifikasi ambang kepercayaan yang tepat untuk kasus penggunaan Anda, dan gunakan ambang kepercayaan tinggi dalam situasi yang membutuhkan akurasi tinggi. Untuk kasus penggunaan tertentu, hasil harus ditinjau dan diverifikasi oleh pengulas manusia yang terlatih dengan tepat. Gunakan Amazon Comprehend Medical dalam skenario perawatan pasien hanya setelah profesional medis terlatih telah meninjau hasil untuk akurasi dan penilaian medis yang baik.

Wilayah yang didukung

Untuk daftar Wilayah AWS tempat Amazon Comprehend Medical tersedia, lihat [Wilayah AWS dan Titik Akhir di Referensi Umum Amazon Web Services](#).

Throttling

[Untuk informasi tentang pembatasan dan kuota Amazon Comprehend Medical, dan untuk meminta peningkatan kuota, lihat AWS Service Quotas.](#)

Kuota keseluruhan

Pengkodean karakter untuk Amazon Comprehend Medical ada di UTF-8. Operasi Amazon Comprehend Medical memiliki kuota berikut untuk transaksi per detik (TPS) atau karakter per detik (CPS):

Sumbu	Default
Trans per detik (TPS) untuk inisialisasi v2, Dete Inf	100 TPS
Inf	tEnt
Inf	Ent
Inf	o

Trans per detik (TPS) untuk Infe EDCT	2 TPS
Karakter per detik (CPS) untuk inisialisasi v2, Dete Inf	40.000 CPS
Inf	tEnt
Inf	Ent
Inf	o

Sumatraya	Default
im , dan opera Infe ØCM	
Karakteristik per detik (CPS untuk Infe EDCT	5000 CPS

Sumit	Default
Trans 5	
per TPS	
detik	
(TPS)	
untuk	Enti
ties	
tion	
„St	ID
etec	
ob ,s	rtit
iesD	
ionV	,StopPHIDe
tect	
b ,St	ID1
0CMI	
nceJ	
Star	
rmIn	
ceJo	pICD10
CMIn	
ceJo	
dan	
Stop	
mInf	
eJob	
opera	

Sumit daya	Default
Trans per detik (TPS) untuk dan Star EDCT nceJ Stop DCTn ceJo	5 TPS

Sumit	Default
Trans per detik (TPS) untuk analisisDionV, „Li tect bs ,I ntit tect Job HIDE onJo CMIn ceJo List mInf eJob CD10 eren , dan Desc xNor renc opera	10 TPS ntit De ibeE ribeP stICD10 cribeI

Sumit daya	Default
Trans per detik (TPS) untuk dan List DCTn ceJo Desc NOME eren	10 TPS

Kuota ukuran untuk file ditunjukkan pada tabel berikut:

Deskripsi	Kuota
Ukuran dokumen maksimum (karakter UTF-8) untuk <code>DetectEntities</code> , <code>DetectEntities-v2</code> , dan <code>DetectPHI</code> operasi.	20 KB
Ukuran dokumen maksimum (karakter UTF-8) untuk <code>InferICD10-CM</code> dan operasi <code>InferRxNorm</code>	10 KB
Analisis Teks ukuran pekerjaan batch maksimum (jumlah total untuk semua file yang dikirimkan dalam pekerjaan batch)	1 GB
Ontologi (ICD10, RX Norm, SNOMEDCT) Menghubungkan ukuran pekerjaan batch maksimum (jumlah total untuk semua file yang dikirimkan dalam pekerjaan batch)	1 GB
Ukuran minimum tugas batch (semua file)	1 byte

Deskripsi	Kuota
InfericD10cm ukuran file individu maksimum untuk pekerjaan batch	40 KB
InferRxNorm ukuran file individu maksimum untuk pekerjaan batch	40 KB
InferSNOMEDCT ukuran file individu maksimum dalam suatu pekerjaan.	5 KB
StartSNOMEDCTInferenceJob ukuran file individu maksimum dalam suatu pekerjaan.	5 KB

Jika teks Anda lebih besar dari kuota karakter, gunakan [segment.py](#) untuk membuat segmen yang lebih kecil yang dapat dianalisis.

Hanya dokumen dalam bahasa Inggris (EN) yang didukung.

Riwayat dokumen untuk Amazon Comprehend Medical

Tabel berikut menjelaskan dokumentasi untuk rilis Amazon Comprehend Medical ini.

Perubahan	Deskripsi	Tanggal
Versi API yang diperbarui untuk InferNomeDCT	Operasi API InferSNOM EDCT sekarang menggunakan an2.6.0.20220301 .	12 Juni 2023
Sifat dan atribut baru ditambahkan untuk DetectEntities V2	Operasi API DetectEntitiesV2 sekarang menggunakan versi2.4.0. Pembaruan ini menambahkan sifat dan atribut baru.	12 Juni 2023
Sifat dan atribut baru ditambahkan untuk DetectEntities	Operasi API DetectEntities sekarang menggunakan an versi0.7.0. Pembaruan ini menambahkan sifat dan atribut baru.	12 Juni 2023
Sifat baru ditambahkan ke InferRxNorm	Operasi API InferRxNorm sekarang menggunakan an versi2.2.0.20221003 . Pembaruan ini menambahkan PAST_HISTORY sifat ke obat yang terdeteksi.	12 Juni 2023
Atribut baru ditambahkan ke InfericD10cm	Operasi API InferICD10CM sekarang menggunakan an versi2.5.0.20220401 . Pembaruan ini menambahkan QUALITY atribut ke kategori kondisi medis.	12 Juni 2023
Versi API yang diperbarui untuk InferRxNorm	Operasi API InferRxNorm sekarang menggunakan	28 April 2023

an versi 2.1.0.20221003 .
Pembaruan ini meningkatkan
kinerja InferRxNorm model.

[Versi API yang diperbarui
untuk InferSNOMEDCT](#)

Operasi API InferSNOM
EDCT sekarang menggunak
an 2.5.0.20220301 .
Pembaruan ini meningkatkan
kinerja model.

April 21, 2023

[Versi API yang diperbarui
untuk InferICD10cm](#)

Operasi API InferICD1
0CM sekarang menggunak
an versi 2.4.0.20220401 .
Pembaruan ini meningkatkan
deteksi DX_NAME entitas dan
DIRECTION atribut.

Maret 17, 2023

[Versi API yang diperbarui
untuk InferSNOMEDCT](#)

Operasi API InferSNOM
EDCT sekarang menggunak
an versi 2.4.0.202
20301 . Pembaruan ini
meningkatkan deteksi
konsep kardiologi dalam
TEST_TREATMENT_PRO
CEDURE kategori MEDICAL_C
ONDITION dan.

15 Maret 2023

[Versi API yang diperbarui
untuk InferICD10cm](#)

Operasi API InferICD1
0CM sekarang menggunak
an versi 2.3.0.20220401 .
Pembaruan ini meningkat
kan deteksi konsep kardiolog
i dalam MEDICAL_C
ONDITION kategori.

15 Maret 2023

[Versi API yang diperbarui untuk DetectEntities V2](#)

Operasi API DetectEntitiesV2 sekarang menggunakan versi 2.3.0. Pembaruan ini meningkatkan deteksi konsep kardiologi dalam TEST_TREATMENT_PROCEDURE kategori MEDICAL_CONDITION dan.

15 Maret 2023

[Versi API yang diperbarui untuk DetectEntities](#)

Operasi API DetectEntities sekarang menggunakan versi 0.6.0. Pembaruan ini meningkatkan deteksi konsep kardiologi dalam TEST_TREATMENT_PROCEDURE kategori MEDICAL_CONDITION dan.

15 Maret 2023

[Perbarui versi API untuk InferNomeDCT](#)

Operasi API InferSNOMEDCT sekarang menggunakan versi 2.3.0.20220301. Pembaruan ini meningkatkan deteksi negasi saat menggunakan operasi api InferNomeDCT.

Februari 8, 2023

[Perbarui versi API untuk InfericD10cm](#)

Operasi API InferICD10CM sekarang menggunakan versi 2.2.0.20220401. Pembaruan ini meningkatkan deteksi negasi saat menggunakan operasi api InfericD10cm.

Februari 8, 2023

[Perbarui versi API untuk DetectEntities V2](#)

Operasi API DetectEntitiesV2 sekarang menggunakan versi 2.2.0. Pembaruan ini meningkatkan deteksi negasi saat menggunakan operasi api DetectEntities V2.

Februari 8, 2023

[Perbarui versi API untuk DetectEntities](#)

Operasi API DetectEntities sekarang menggunakan versi 0.5.0. Pembaruan ini meningkatkan deteksi negasi saat menggunakan operasi DetectEntities api.

Februari 8, 2023

[Perbarui versi API untuk InferenceDCT](#)

Operasi API InferenceDCT sekarang menggunakan versi 2.2.0.20220301. Pembaruan ini memperbaiki bug yang mencegah total set atribut yang diidentifikasi dikembalikan dalam output API.

Desember 9, 2022

[Versi API yang diperbarui untuk InferenceDCT](#)

Operasi API InferenceDCT sekarang menggunakan versi 2.1.0.20220301. Pembaruan ini mencakup peningkatan akurasi, dan sifat baru terdeteksi.

16 November 2022

<u>Versi API yang diperbarui untuk InferRxNorm</u>	Operasi API InferRxNorm sekarang menggunakan versi 2.1.0.20221003. Pembaruan ini memperbaiki bug yang mencegah total set atribut yang diidentifikasi dikembalikan.	16 November 2022
<u>Versi API yang diperbarui untuk InfericD10cm</u>	Operasi API InferICD10CM sekarang menggunakan versi 2.1.0.20220401. Pembaruan ini mencakup peningkatan akurasi, dan sifat baru terdeteksi.	16 November 2022
<u>Versi API yang diperbarui untuk DetectEntities V2</u>	Operasi API DetectEntitiesV2 sekarang menggunakan versi 2.1.0. Pembaruan ini mencakup peningkatan akurasi. Ini juga menambahkan entitas baru, atribut, hubungan, sifat, dan kategori baru.	16 November 2022
<u>Versi API yang diperbarui untuk DetectEntities</u>	Operasi API DetectEntities sekarang menggunakan versi 0.4.0. Pembaruan ini mencakup peningkatan akurasi, dan sifat baru terdeteksi.	16 November 2022
<u>Batas yang diperbarui untuk StartSNOMEDCTInferenceJob dan InferSNOMEDCT</u>	Ukuran file individual maksimum StartSNOMEDCTInferenceJob dan operasi InferSNOMEDCT API sekarang 5 KB.	September 22, 2022

Pembaruan pemrosesan teks	Amazon Comprehend Medical sekarang menyediakan dukungan yang lebih baik untuk mengurai ruang putih yang tidak relevan dengan benar dan pengembalian carriage di semua operasi API analisis teks.	12 September 2022
Pembaruan pemrosesan teks	Amazon Comprehend Medical sekarang memberikan dukungan yang lebih baik untuk mengurai ruang putih yang tidak relevan dengan benar dan pengembalian carriage di semua operasi API ontologi.	12 September 2022
Versi API yang diperbarui untuk InferNomeDCT	InferSNOWMEDCT Operasi Amazon Comprehend Medical API sekarang menggunakan versi 1.1.0.20220301.	8 Juli 2022
Versi API yang diperbarui untuk InferICD10cm	InferICD10CM Operasi Amazon Comprehend Medical API sekarang menggunakan versi 1.1.0.20220401.	8 Juli 2022
Versi API yang diperbarui untuk InferRxNorm pengoperasian di Amazon Comprehend Medical	Amazon Comprehend Medical sekarang menggunakan RxNorm RxTerms 2022-03-07 dan Rilis untuk setiap RxCui.	Juni 29, 2022

Model yang diperbarui untuk Amazon Comprehend Medical	Amazon Comprehend Medical sekarang menyediakan dukungan yang lebih baik NEGATION untuk deteksi sifat menggunakan Detect Entities V2.	April 25, 2022
Fitur baru untuk Amazon Comprehend Medical	Anda sekarang dapat menggunakan Amazon Comprehend Medical untuk membuat koneksi pribadi dengan virtual private cloud (VPC) Anda dengan membuat antarmuka VPC endpoint. Untuk informasi selengkapnya, lihat titik akhir VPC (). PrivateLink	13 Juni 2021
Fitur baru untuk Amazon Comprehend Medical	Amazon Comprehend Medical sekarang menyediakan operasi batch untuk penautan ontologi. Ini memungkinkan layanan untuk mendeteksi entitas dalam teks medis yang disimpan dalam bucket S3 dan menautkan entitas tersebut ke ontologi standar. Untuk informasi lebih lanjut, lihat Ontology Linking Batch Analysis .	4 Mei 2020

[Fitur baru untuk Amazon Comprehend Medical](#)

Anda sekarang dapat menggunakan Amazon Comprehend Medical untuk mengekstrak dan menghubungkan ekspresi tanggal atau waktu ke salah satu entitas yang saat ini terdeteksi oleh Amazon Comprehend Medical. Untuk informasi selengkapnya, lihat [Mendeteksi Entitas Versi 2](#).

2 Maret 2020

[Fitur baru](#)

Amazon Comprehend Medical sekarang mendeteksi obat-obatan dan kondisi medis dan menghubungkan entitas dengan ontologi nasional dan ICD-10-CM yang mapan. RxNorm Untuk informasi selengkapnya, lihat [Ontology Linking API](#).

16 Desember 2019

[Fitur baru](#)

Amazon Comprehend Medical sekarang menyediakan operasi batch sehingga Anda dapat memproses teks medis yang disimpan dalam ember S3. Ini juga menyediakan model baru yang dapat Anda gunakan untuk memeriksa teks medis Anda. Untuk informasi selengkapnya, lihat [Mendeteksi Entitas Versi 2](#).

24 September 2019

Fitur baru utama

Amazon Comprehend Medical 27 November 2018 adalah layanan baru yang mendeteksi informasi berguna dalam teks klinis yang tidak terstruktur: catatan dokter, ringkasan pelepasan, hasil tes, catatan kasus, dan sebagainya.

a. Amazon Comprehend Medical menggunakan model pemrosesan bahasa alami (NLP) untuk memanfaatkan kemajuan terbaru dalam pembelajaran mesin untuk memilah-milah jumlah data yang sangat besar ini dan mengambil informasi berharga yang sulit diambil dan digunakan tanpa upaya manual yang signifikan. Untuk informasi lebih lanjut, lihat [Amazon Comprehend Medical](#).

Amazon Comprehend Medical log perubahan

Bagian berikut merinci pembaruan ke layanan Amazon Comprehend Medical.

Perubahan pada operasi Amazon Comprehend Medical API

Tanggal rilis: 2023-06-12

Tipe: Otomatis

Dalam pembaruan ini, operasi API berikut sekarang menggunakan versi model baru.

- DetectEntities sekarang menggunakan model versi 0.7.0.
- DetectEntitiesV2 sekarang menggunakan model versi 2.4.0.
- InferRxNorm sekarang menggunakan model versi 2.2.0.20221003.

- InfericD10cm sekarang menggunakan model versi 2.5.0.20220401.
- InfersNomeDCT sekarang menggunakan model versi 2.6.0.20220301.

Pembaruan ini menambahkan sifat PAST_HISTORY ke kategori obat dalam operasiDetectEntities, InferRxNorm dan DetectEntitiesV2 API.

Pembaruan ini juga menambahkan atribut QUALITY ke kategori kondisi medis diDetectEntitiesV2, DetectEntities, dan InferICD10CM.

Untuk mempelajari selengkapnya tentang operasi API analisis teks Amazon Comprehend Medical, lihat. [Operasi](#)

Untuk mempelajari lebih lanjut tentang ontologi Amazon Comprehend Medical yang menghubungkan operasi API, lihat. [Ontologi menghubungkan](#)

Perubahan pada operasi Amazon Comprehend Medical API

Tanggal rilis: 2023-04-28

Tipe: Otomatis

Dalam pembaruan ini, operasi InferRxNorm API diperbarui. Operasi InferRxNorm API sekarang menggunakan model versi 2.1.0.20221003. Pembaruan ini meningkatkan kinerja model.

Perubahan pada operasi Amazon Comprehend Medical API

Tanggal rilis: 2023-04-21

Tipe: Otomatis

Dalam pembaruan ini, operasi API InfersNomeDCT telah diperbarui. Operasi API InfersNomeDCT sekarang menggunakan model versi 2.5.0.20220301. Pembaruan ini meningkatkan kinerja model.

Perubahan pada operasi Amazon Comprehend Medical API

Tanggal rilis: 2023-03-17

Tipe: Otomatis

Dalam pembaruan ini, operasi API InfericD10cm telah diperbarui. Pembaruan ini meningkatkan deteksi DX_NAME entitas dan DIRECTION atribut. Itu juga membuat penyesuaian pemrosesan sistem kecil yang memengaruhi deteksi sifat dan atribut dan skor frekuensi/kepercayaan dalam output API.

Perubahan pada operasi Amazon Comprehend Medical API

Tanggal rilis: 2023-03-15

Tipe: Otomatis

Dalam pembaruan ini, operasi Amazon Comprehend Medical API berikut diperbarui `InferSNOMEDCT::,` dan `InferICD10CM DetectEntities DetectEntitiesV2`. Pembaruan ini otomatis, dan meningkatkan deteksi kondisi medis, tes, perawatan, dan prosedur, bersama dengan atribut dan sifat terkait, dalam spesialisasi kardiologi.

Perubahan pada operasi Amazon Comprehend Medical API

Tanggal rilis: 2023-02-08

Tipe: Otomatis

Dalam pembaruan ini, operasi Amazon Comprehend Medical API berikut diperbarui `InferSNOMEDCT::,` dan `InferICD10CM DetectEntities DetectEntitiesV2`. Pembaruan ini otomatis, dan meningkatkan deteksi negasi dalam kondisi medis saat memindai teks klinis.

Perubahan pada operasi API Amazon Comprehend Medical InferNomeDCT

Tanggal rilis: 2022-12-09

Tipe: Otomatis

Operasi Amazon Comprehend Medical API `InferSNOMEDCT`, diperbarui untuk menggunakan versi `2.2.0.20220301`. Pembaruan ini memperbaiki bug yang mencegah total set atribut yang diidentifikasi dikembalikan.

Analisis Batch yang dilakukan menggunakan operasi `StartSNOMEDCTInferenceJob` API menggunakan versi API yang sama, `2.2.0.20220301`. Untuk mempelajari selengkapnya tentang menggunakan operasi API ini, lihat [Menghubungkan CT SNOMED](#).

Perubahan pada operasi Amazon Comprehend Medical API

Tanggal rilis: 2022-11-16

Tipe: Otomatis

Dalam pembaruan ini, operasi Amazon Comprehend Medical API berikut telah diperbarui:,,, InferSNOMEDCT InferICD10CM DetectEntities DetectEntitiesV2

DetectEntities

DetectEntities Operasi Amazon Comprehend Medical API sekarang menggunakan API versi 0.4.0. Pembaruan ini mencakup peningkatan akurasi, dan sifat baru terdeteksi.

Untuk Kategori:TEST_TREATMENT_PROCEDURE, Jenis:PROCEDURE_NAME, ciri-ciri berikut ditambahkan.

- NEGATION
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

Untuk Kategori:TEST_TREATMENT_PROCEDURE, Jenis:TEST_NAME, ciri-ciri berikut ditambahkan.

- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

Untuk Kategori:TEST_TREATMENT_PROCEDURE, Jenis:TREATMENT_NAME, ciri-ciri berikut ditambahkan.

- NEGASI
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

Untuk Kategori:MEDICAL_CONDITION, Jenis:DX_NAME, ciri-ciri berikut ditambahkan.

- HYPOTHETICAL
- LOW_CONFIDENCE
- PERTAINS_TO_FAMILY

DetectEntitiesV2

DetectEntitiesV2 Operasi Amazon Comprehend Medical API sekarang menggunakan API versi 2.1.0. Pembaruan mencakup kategori baru BEHAVIORAL_ENVIRONMENTAL_SOCIAL bersama dengan pembaruan untuk kategori dan sifat yang ada.

Untuk Kategori baru:BEHAVIORAL_ENVIRONMENTAL_SOCIAL,
jenis berikut telah ditambahkan:GENDER,,RACE_ETHNICITY,
ALLERGIES,TOBACCO_USE,ALCOHOL_CONSUMPTION,REC_DRUG_USE,UnmappedAttributes.

Untuk Tipe: GENDER

- Tidak ada atribut atau sifat yang tersedia.

Untuk Tipe: RACE_ETHNICITY

- Tidak ada atribut atau sifat yang tersedia.

Untuk Tipe:ALLERGIES, ciri-ciri berikut ditambahkan.

- NEGATION
- PAST_HISTORY

Untuk Type:TOBACCO_USE, atribut dan sifat berikut ditambahkan.

- Atribut
 - AMOUNT
 - DURATION
 - FREQUENCY
- Sifat
 - NEGATION
 - PAST_HISTORY

Untuk Type:ALCOHOL_CONSUMPTION, atribut dan sifat berikut ditambahkan.

- Atribut
 - AMOUNT
 - DURATION
 - FREQUENCY
- Sifat
 - NEGATION
 - PAST_HISTORY

Untuk Type:REC_DRUG_USE, atribut dan sifat berikut ditambahkan.

- Atribut

- AMOUNT
- DURATION
- FREQUENCY
- Sifat
 - NEGATION
 - PAST_HISTORY

Untuk Type:UnmappedAttributes, atribut berikut ditambahkan.

- AMOUNT

Untuk Kategori:MEDICAL_CONDITION, Jenis:DX_NAME, ciri-ciri berikut ditambahkan.

- PERTAINS_TO_FAMILY
- HYPOTHETICAL
- LOW_CONFIDENCE

Untuk Kategori:TEST_TREATMENT_PROCEDURE, Jenis:PROCEDURE_NAME, ciri-ciri berikut ditambahkan.

- NEGATION
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

Untuk Kategori:TEST_TREATMENT_PROCEDURE, Jenis:TEST_NAME, ciri-ciri berikut ditambahkan.

- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

Untuk Kategori:TEST_TREATMENT_PROCEDURE, Jenis:TREATMENT_NAME, ciri-ciri berikut ditambahkan.

- NEGATION
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

Analisis Batch yang dilakukan menggunakan operasi `StartEntitiesDetectionV2Job` API menggunakan versi API yang sama, 2.1.0. Untuk mempelajari selengkapnya tentang menggunakan operasi API ini, lihat [Mendeteksi entitas \(Versi 2\)](#).

InferSNOMEDCT

`InferSNOMEDCT` Operasi Amazon Comprehend Medical API sekarang menggunakan API versi 2.1.0.20220301. Pembaruan versi ini menambahkan sifat baru ke kategori dan jenis berikut.

Untuk Kategori: `MEDICAL_CONDITION`, Jenis: `DX_NAME`, ciri-ciri berikut ditambahkan.

- `PERTAINS_TO_FAMILY`
- `HYPOTHETICAL`
- `LOW_CONFIDENCE`

Untuk Kategori: `TEST_TREATMENT_PROCEDURE`, Jenis: `TEST_NAME`, ciri-ciri berikut ditambahkan.

- `PAST_HISTORY`
- `FUTURE`
- `HYPOTHETICAL`

Untuk Kategori: `TEST_TREATMENT_PROCEDURE`, Jenis: `TREATMENT_NAME`, ciri-ciri berikut ditambahkan.

- `NEGATION`
- `PAST_HISTORY`
- `FUTURE`
- `HYPOTHETICAL`

Perbaikan bug

- Pembaruan ini memperbaiki bug yang mencegah total set atribut yang diidentifikasi dikembalikan.

Analisis Batch yang dilakukan menggunakan operasi `StartSNOMEDCTInferenceJob` API menggunakan versi yang sama. Untuk mempelajari selengkapnya tentang menggunakan operasi API ini, lihat [Menghubungkan CT SNOMED](#).

InferICD10CM

`InferICD10CM` Operasi Amazon Comprehend Medical API sekarang menggunakan API versi 2.1.0.20220401. Pembaruan versi ini menambahkan sifat baru ke kategori dan jenis berikut.

Untuk Kategori: MEDICAL_CONDITION, Jenis: DX_NAME, ciri-ciri berikut ditambahkan.

- PERTAINS_TO_FAMILY
- HYPOTHETICAL
- LOW_CONFIDENCE

Perbaikan bug

- Pembaruan ini memperbaiki bug yang mencegah total set atribut yang diidentifikasi dikembalikan.

Analisis Batch yang dilakukan menggunakan operasi StartRxNormInferenceJob API menggunakan versi yang sama. Untuk mempelajari selengkapnya tentang penggunaan operasi InferRxNorm dan StartRxNormInferenceJob API, lihat [RxNorm menghubungkan](#).

InferRxNorm

Operasi Amazon Comprehend Medical API sekarang menggunakan versi API InferRxNorm .
1.3.1.20221003

Perbaikan bug

- Pembaruan ini memperbaiki bug yang mencegah total set atribut yang diidentifikasi dikembalikan.

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.