aws

Panduan Pengguna

Amazon Fraud Detector

Versi latest

Amazon Fraud Detector: Panduan Pengguna

Table of Contents

Apa itu Amazon Fraud Detector?

Amazon Fraud Detector adalah layanan deteksi penipuan yang dikelola sepenuhnya yang mengotomatiskan deteksi aktivitas yang berpotensi penipuan secara online. Kegiatan ini termasuk transaksi yang tidak sah dan pembuatan akun palsu. Amazon Fraud Detector bekerja dengan menggunakan pembelajaran mesin untuk menganalisis data Anda. Ini dilakukan dengan cara yang dibangun dari keahlian berpengalaman selama lebih dari 20 tahun deteksi penipuan di Amazon.

Anda dapat menggunakan Amazon Fraud Detector untuk membuat model deteksi penipuan yang disesuaikan, menambahkan logika keputusan untuk menafsirkan evaluasi penipuan model, dan menetapkan hasil seperti lulus atau kirim untuk ditinjau untuk setiap kemungkinan evaluasi penipuan. Dengan Amazon Fraud Detector, Anda tidak memerlukan keahlian pembelajaran mesin untuk mendeteksi aktivitas penipuan.

Untuk memulai, kumpulkan dan siapkan data penipuan yang Anda kumpulkan di organisasi Anda. Amazon Fraud Detector kemudian menggunakan data ini untuk melatih, menguji, dan menerapkan model deteksi penipuan khusus atas nama Anda. Sebagai bagian dari proses ini, Amazon Fraud Detector menggunakan model pembelajaran mesin yang telah mempelajari pola penipuan AWS dan keahlian penipuan Amazon sendiri untuk mengevaluasi data penipuan Anda dan menghasilkan skor model dan data kinerja model. Anda mengonfigurasi logika keputusan untuk menafsirkan skor model dan menetapkan hasil untuk cara menangani setiap evaluasi penipuan.

Manfaat

Amazon Fraud Detector memberikan manfaat berikut. Manfaat ini memungkinkan Anda mendeteksi penipuan dengan cepat tanpa perlu menginvestasikan waktu dan sumber daya yang secara tradisional diperlukan untuk membangun dan memelihara sistem manajemen penipuan.

Pembuatan model penipuan otomatis

Model deteksi penipuan Amazon Fraud Detector adalah model pembelajaran mesin otomatis yang disesuaikan untuk memenuhi kebutuhan bisnis spesifik Anda. Anda dapat menggunakan model Amazon Fraud Detector untuk mengidentifikasi potensi penipuan dalam transaksi online apa pun seperti pembuatan akun baru, pembayaran online, dan checkout tamu.

Karena model penipuan dibuat melalui proses otomatis, Anda dapat melupakan banyak langkah yang terkait dengan pembuatan dan pelatihan model. Langkah-langkah ini termasuk validasi dan

pengayaan data, rekayasa fitur, pemilihan algoritme, penyetelan hiperparameter, dan penerapan model.

Untuk membuat model deteksi penipuan menggunakan Amazon Fraud Detector, Anda hanya mengunggah kumpulan data penipuan historis perusahaan Anda dan memilih jenis model. Kemudian, Amazon Fraud Detector secara otomatis menemukan algoritme deteksi penipuan yang paling sesuai untuk kasus penggunaan Anda dan membuat modelnya. Anda tidak perlu tahu coding atau memiliki keahlian pembelajaran mesin untuk membuat model deteksi penipuan.

Model penipuan yang berkembang dan belajar

Model deteksi penipuan harus terus berkembang untuk mengikuti lanskap penipuan yang berubah. Amazon Fraud Detector melakukan ini secara otomatis dengan menghitung informasi termasuk usia akun, waktu sejak aktivitas terakhir, dan jumlah aktivitas. Hasilnya adalah model Anda mempelajari perbedaan antara pelanggan tepercaya yang sering melakukan transaksi dan upaya lanjutan yang khas dari penipu. Ini membantu mempertahankan kinerja model Anda lebih lama di antara sesi pelatihan ulang.

Visualisasi kinerja model penipuan

Setelah model Anda dilatih menggunakan data yang Anda berikan, Amazon Fraud Detector memvalidasi performa model Anda. Ini juga menyediakan alat visual bagi Anda untuk menilai kinerja. Untuk setiap model yang Anda latih, Anda dapat melihat skor kinerja model, grafik distribusi skor, matriks kebingungan, tabel ambang batas, dan semua input yang Anda berikan diberi peringkat berdasarkan dampaknya terhadap kinerja model. Dengan menggunakan alat kinerja ini, Anda dapat mempelajari kinerja model Anda dan input apa yang mendorong kinerja model Anda. Jika diperlukan, Anda dapat mengubah model Anda untuk meningkatkan kinerjanya secara keseluruhan.

Prediksi penipuan

Amazon Fraud Detector menghasilkan prediksi penipuan untuk aktivitas bisnis organisasi Anda. Prediksi penipuan adalah evaluasi aktivitas bisnis untuk risiko penipuan. Amazon Fraud Detector menghasilkan prediksi menggunakan logika prediksi dengan data yang terkait dengan aktivitas. Anda memberikan data ini ketika Anda membuat model deteksi penipuan Anda. Anda bisa mendapatkan prediksi penipuan untuk satu aktivitas secara real time atau mendapatkan prediksi penipuan offline untuk serangkaian aktivitas.

Visualisasi penjelasan prediksi penipuan

Amazon Fraud Detector menghasilkan penjelasan prediksi sebagai bagian dari proses prediksi penipuan. Penjelasan prediksi memberikan wawasan tentang bagaimana setiap elemen data yang

digunakan untuk melatih model Anda telah memengaruhi skor prediksi penipuan model Anda. Penjelasan prediksi disediakan dengan menggunakan alat visual seperti tabel dan grafik. Anda dapat menggunakan alat ini untuk mengidentifikasi secara visual seberapa besar pengaruh setiap elemen data terhadap skor prediksi. Kemudian, Anda dapat menggunakan informasi ini untuk menganalisis pola penipuan di seluruh kumpulan data Anda dan mendeteksi bias, jika ada. Terakhir Anda juga dapat menggunakan penjelasan prediksi untuk mengidentifikasi indikator risiko teratas selama proses investigasi penipuan manual. Ini membantu Anda mempersempit akar penyebab yang mengarah pada prediksi positif palsu.

Tindakan berbasis aturan

Setelah model deteksi penipuan Anda dilatih, Anda dapat menambahkan aturan untuk mengambil tindakan pada data yang dievaluasi, seperti menerima data, mengirim data untuk ditinjau, atau mengumpulkan lebih banyak data. Aturan adalah kondisi yang memberi tahu Amazon Fraud Detector bagaimana menafsirkan data selama prediksi penipuan. Misalnya, Anda dapat membuat aturan yang menandai akun pelanggan yang mencurigakan untuk ditinjau. Anda dapat mengatur aturan ini untuk dimulai jika kedua skor model yang terdeteksi lebih besar dari ambang batas yang telah ditentukan sebelumnya dan jika kode otorisasi pembayaran akun (AUTH_CODE) tidak valid.

Konsep dan istilah inti

Berikut ini adalah daftar konsep inti dan istilah yang digunakan dalam Amazon Fraud Detector:

Peristiwa

Acara adalah aktivitas bisnis organisasi Anda yang dievaluasi untuk risiko penipuan. Amazon Fraud Detector menghasilkan prediksi penipuan untuk acara.

Label

Label mengklasifikasikan satu peristiwa sebagai penipuan atau sah. Label digunakan untuk melatih model pembelajaran mesin di Amazon Fraud Detector.

Entitas

Entitas mewakili siapa yang melakukan peristiwa tersebut. Anda memberikan ID entitas sebagai bagian dari data penipuan perusahaan Anda untuk menunjukkan entitas tertentu yang melakukan acara tersebut.

Jenis peristiwa

Jenis acara mendefinisikan struktur bagi acara yang dikirimkan ke Amazon Fraud Detector. Ini termasuk data yang dikirim sebagai bagian dari acara, entitas yang melakukan acara (seperti pelanggan), dan label yang mengklasifikasikan acara. Contoh jenis acara termasuk transaksi pembayaran online, pendaftaran akun, dan otentikasi.

Jenis entitas

Jenis entitas mengklasifikasikan entitas. Contoh klasifikasi termasuk pelanggan, pedagang, atau akun.

Dataset acara

Dataset acara adalah data historis perusahaan Anda tentang aktivitas bisnis atau acara tertentu. Misalnya, acara perusahaan Anda mungkin pendaftaran akun online. Data dari satu peristiwa (pendaftaran) mungkin termasuk alamat IP terkait, alamat email, alamat penagihan, dan stempel waktu acara. Anda menyediakan kumpulan data peristiwa ke Amazon Fraud Detector untuk membuat dan melatih model deteksi penipuan.

Model

Model adalah output dari algoritma pembelajaran mesin. Algoritma ini diimplementasikan dalam kode dan dijalankan pada data peristiwa yang Anda berikan.

Jenis model

Jenis model mendefinisikan algoritma, pengayaan, dan transformasi fitur yang digunakan selama pelatihan model. Ini juga mendefinisikan persyaratan data untuk melatih model. Definisi ini berfungsi untuk mengoptimalkan model Anda untuk jenis penipuan tertentu. Anda menentukan jenis model yang akan digunakan saat Anda membuat model Anda.

Pelatihan model

Pelatihan model adalah proses menggunakan dataset acara yang disediakan untuk membuat model yang dapat memprediksi peristiwa penipuan. Semua langkah dalam proses pelatihan model sepenuhnya otomatis. Langkah-langkah ini termasuk validasi data, transformasi data, rekayasa fitur, pemilihan algoritme, dan pengoptimalan model.

Skor model

Skor model adalah hasil evaluasi data penipuan historis perusahaan Anda. Selama proses pelatihan model, Amazon Fraud Detector mengevaluasi kumpulan data untuk aktivitas penipuan

dan menghasilkan skor antara 0 dan 1000. Untuk skor ini, 0 mewakili risiko penipuan rendah sedangkan 1000 mewakili risiko penipuan tertinggi. Skor itu sendiri terkait langsung dengan tingkat positif palsu (FPR).

Versi model

Versi model adalah output dari pelatihan model.

Penyebaran model

Penyebaran model adalah proses untuk mengaktifkan versi model dan membuatnya tersedia untuk menghasilkan prediksi penipuan.

Titik akhir SageMaker model Amazon

Selain membuat model menggunakan Amazon Fraud Detector, Anda dapat menggunakan endpoint model yang SageMaker di-host secara opsional dalam evaluasi Amazon Fraud Detector.

Untuk informasi selengkapnya tentang membuat model SageMaker, lihat Melatih Model dengan Amazon SageMaker.

Detektor

Detektor berisi logika deteksi seperti model dan aturan untuk peristiwa tertentu yang ingin Anda evaluasi untuk penipuan. Anda membuat detektor menggunakan versi model.

Versi detektor

Detektor dapat memiliki beberapa versi, dengan setiap versi memiliki status`Draft`,`Active`, atau`Inactive`. Hanya satu versi detektor yang dapat `Active` berstatus sekaligus.

Variabel

Variabel mewakili elemen data yang terkait dengan peristiwa yang ingin Anda gunakan dalam prediksi penipuan. Variabel dapat dikirim dengan peristiwa sebagai bagian dari prediksi penipuan atau turunan, seperti output dari model Amazon Fraud Detector atauAmazon SageMaker.

Aturan

Aturan adalah kondisi yang memberi tahu Amazon Fraud Detector bagaimana menafsirkan nilai variabel selama prediksi penipuan. Aturan terdiri dari satu atau lebih variabel, ekspresi logika, dan satu atau lebih hasil. Variabel yang digunakan dalam aturan harus menjadi bagian dari dataset peristiwa yang dievaluasi detektor. Selain itu setiap detektor harus memiliki setidaknya satu aturan yang terkait dengannya.

Hasil

Ini adalah hasil, atau output, dari prediksi penipuan. Setiap aturan yang digunakan dalam prediksi penipuan harus menentukan satu atau lebih hasil.

Prediksi penipuan

Prediksi penipuan adalah evaluasi penipuan baik untuk satu peristiwa atau serangkaian peristiwa. Amazon Fraud Detector menghasilkan prediksi penipuan untuk satu acara online secara real time dengan secara sinkron memberikan skor model dan hasil berdasarkan aturan. Amazon Fraud Detector menghasilkan prediksi penipuan untuk serangkaian acara offline. Anda dapat menggunakan prediksi untuk melakukan offline proof-of-concept, atau untuk mengevaluasi risiko penipuan secara retrospektif setiap jam, harian, atau mingguan.

Penjelasan prediksi penipuan

Penjelasan prediksi penipuan memberikan wawasan tentang bagaimana setiap variabel memengaruhi skor prediksi penipuan model Anda. Ini memberikan informasi tentang bagaimana setiap variabel mempengaruhi skor risiko dalam hal besarnya (mulai dari 0 hingga 5 dengan 5 tertinggi) dan arah (mendorong skor lebih tinggi atau lebih rendah).

Bagaimana Amazon Fraud Detector bekerja

Amazon Fraud Detector membuat model pembelajaran mesin yang disesuaikan untuk mendeteksi potensi aktivitas online penipuan dalam bisnis Anda. Untuk memulai, Anda menyediakan kasus penggunaan bisnis Anda. Bergantung pada kasus penggunaan bisnis Anda, Amazon Fraud Detector merekomendasikan jenis model yang akan digunakan untuk membuat model deteksi penipuan untuk Anda. Selain itu, ini juga memberikan wawasan tentang elemen data yang perlu Anda berikan sebagai bagian dari data historis bisnis Anda. Amazon Fraud Detector menggunakan kumpulan data historis untuk secara otomatis membuat dan melatih model yang disesuaikan untuk Anda.

Proses pelatihan model otomatis melibatkan pemilihan algoritma pembelajaran mesin yang mendeteksi penipuan untuk kasus penggunaan bisnis spesifik Anda, memvalidasi data yang Anda berikan, dan melakukan manipulasi data untuk meningkatkan kinerja model. Setelah melatih model, Amazon Fraud Detector menghasilkan skor model dan metrik kinerja model lainnya. Anda dapat menggunakan skor dan metrik kinerja untuk mengevaluasi kinerja model. Jika perlu, Anda dapat menambah atau menghapus elemen data dari kumpulan data yang Anda berikan untuk pelatihan dan melatih kembali model untuk meningkatkan skor model.

Setelah model dibuat, dilatih, dan diaktifkan, Anda perlu mengonfigurasi logika keputusan, juga dikenal sebagai aturan, yang memberi tahu model bagaimana menafsirkan data yang dihasilkan oleh bisnis Anda, dan menetapkan hasil untuk bagaimana menangani interpretasi setiap aktivitas. Hasil dapat mewakili tindakan seperti, menyetujui atau meninjau aktivitas, atau dapat mewakili tingkat risiko aktivitas seperti risiko tinggi, risiko sedang, dan risiko rendah.

Sebuah detektor adalah kontainer yang menyimpan model Anda dan aturan yang terkait. Anda perlu membuat, menguji, dan menyebarkan detektor ke lingkungan produksi Anda.

Detektor yang digunakan di lingkungan produksi Anda menyediakan kemampuan deteksi penipuan untuk aplikasi bisnis Anda. Untuk melakukan evaluasi penipuan, model membandingkan semua data yang masuk dari aktivitas bisnis Anda dengan data historis bisnis Anda dan menggunakan algoritme pembelajaran mesin canggih dengan aturan yang Anda buat untuk menganalisis hasil dan menetapkan hasil. Dengan Amazon Fraud Detector, Anda dapat mengevaluasi data dari satu aktivitas bisnis secara real-time atau mengevaluasi data dari berbagai aktivitas bisnis secara offline.

Katakanlah Anda memiliki bisnis yang memiliki transfer dana online sebagai salah satu kegiatannya. Anda ingin menggunakan Amazon Fraud Detector untuk mendeteksi permintaan penipuan untuk transfer dana, secara real time. Untuk memulai, Anda harus terlebih dahulu memberikan Amazon Fraud Detector dengan data dari permintaan transfer dana sebelumnya. Amazon Fraud Detector menggunakan data ini untuk membuat dan melatih model yang disesuaikan untuk mendeteksi permintaan penipuan untuk transfer dana. Dan kemudian, Anda membuat detektor dengan menambahkan model dan dengan mengonfigurasi aturan untuk model Anda untuk menafsirkan data. Contoh aturan untuk aktivitas transfer dana online dapat berupa, jika permintaan transfer dana berasal darixyz@example.comalamat email, kirim permintaan untuk ditinjau. Dalam lingkungan produksi bisnis Anda, ketika permintaan transfer dana masuk, model menganalisis data yang datang dengan permintaan dan menggunakan aturan untuk menetapkan hasilnya. Anda kemudian dapat mengambil tindakan atas permintaan tergantung pada hasil yang ditetapkan.

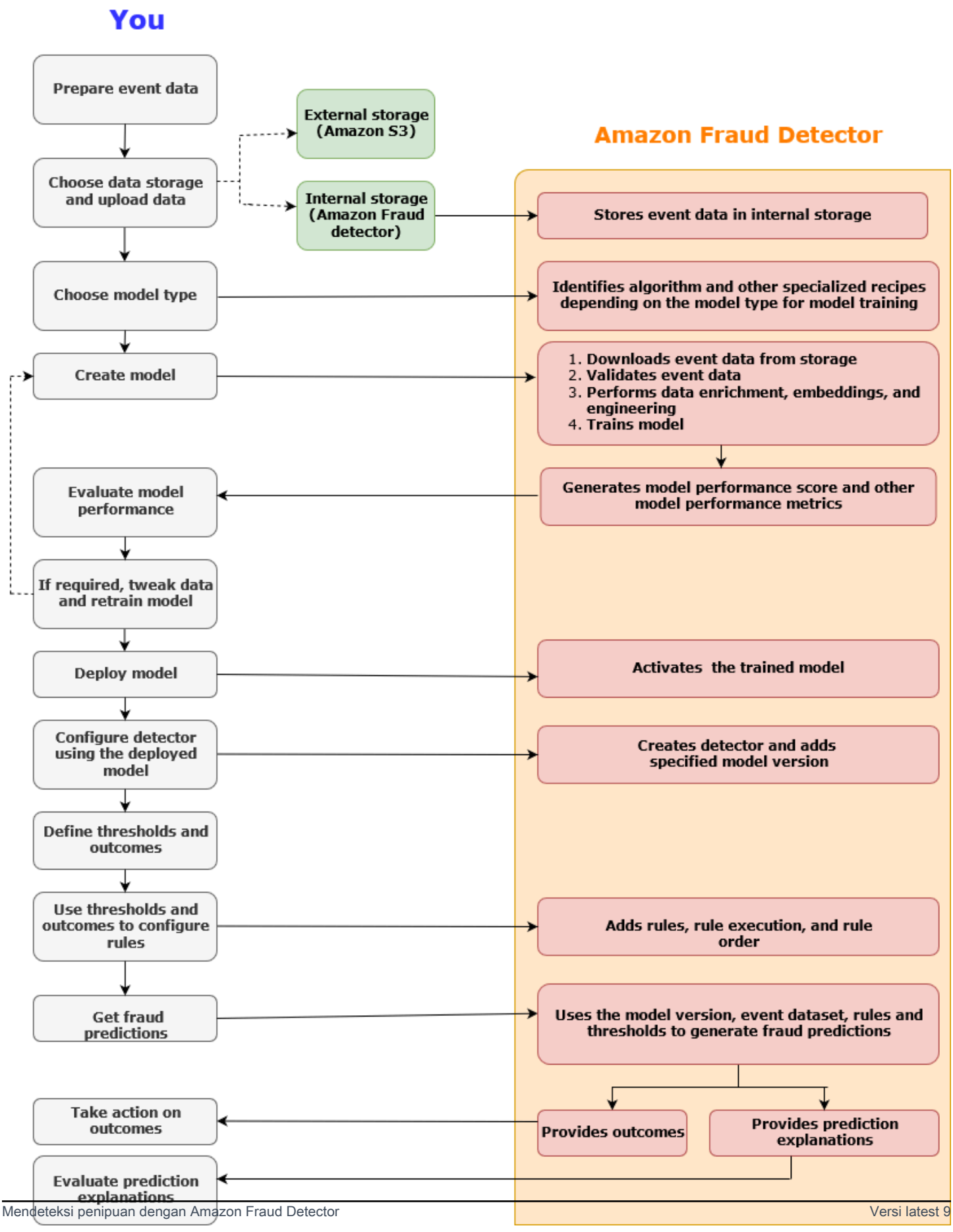
Amazon Fraud Detector menggunakan komponen seperti, kumpulan data pelatihan, model, detektor, aturan, dan hasil untuk memberikan logika evaluasi penipuan kepada bisnis Anda.

Untuk informasi tentang alur kerja yang akan Anda gunakan untuk mendeteksi penipuan menggunakan Amazon Fraud Detector, lihatMendeteksi penipuan dengan Amazon Fraud Detector

Mendeteksi penipuan dengan Amazon Fraud Detector

Bagian ini menjelaskan alur kerja tipikal untuk mendeteksi penipuan dengan Amazon Fraud Detector. Ini juga merangkum bagaimana Anda dapat menyelesaikan tugas-tugas itu. Diagram berikut

memberikan tampilan alur kerja tingkat tinggi untuk mendeteksi penipuan dengan Amazon Fraud Detector.


You
Prepare event data
Choose data storage and upload data
External storage (Amazon S3)
Internal storage (Amazon Fraud detector)
Choose model type
Create model
Evaluate model performance
If required, tweak data and retrain model
Deploy model
Configure detector using the deployed model
Define thresholds and outcomes
Use thresholds and outcomes to configure rules
Get fraud predictions
Take action on outcomes
Evaluate prediction explanations
Amazon Fraud Detector
Stores event data in internal storage
Identifies algorithm and other specialized recipes depending on the model type for model training
1. Downloads event data from storage
2. Validates event data
3. Performs data enrichment, embeddings, and engineering
4. Trains model
Generates model performance score and other model performance metrics
Activates the trained model
Creates detector and adds specified model version
Adds rules, rule execution, and rule order
Uses the model version, event dataset, rules and thresholds to generate fraud predictions
Provides outcomes
Provides prediction explanations

Deteksi penipuan adalah proses yang berkelanjutan. Setelah Anda menerapkan model Anda, pastikan untuk mengevaluasi skor kinerja dan metrik berdasarkan penjelasan prediksi. Dengan demikian, Anda dapat mengidentifikasi indikator risiko teratas, mempersempit akar penyebab yang mengarah ke positif palsu, dan menganalisis pola penipuan di seluruh kumpulan data Anda dan mendeteksi bias, jika ada. Untuk meningkatkan akurasi prediksi, Anda dapat mengubah kumpulan data Anda untuk menyertakan data baru atau yang direvisi. Kemudian, Anda dapat melatih kembali model Anda dengan dataset yang diperbarui. Saat semakin banyak data tersedia, Anda terus melatih ulang model Anda untuk meningkatkan akurasi.

Mengakses Amazon Fraud Detector

Amazon Fraud Detector tersedia dalam beberapa Wilayah AWS dan dapat diakses menggunakan AWS antarmuka.

Ketersediaan

Amazon Fraud Detector tersedia di AS Timur (Virginia N.), AS Timur (Ohio), AS Barat (Oregon), Eropa (Irlandia), Asia Pasifik (Singapura), dan Asia Pasifik (Sydney). Wilayah AWS

Antarmuka

Anda dapat membuat, melatih, menyebarkan, menguji, menjalankan, dan mengelola model dan detektor deteksi penipuan menggunakan salah satu antarmuka berikut:

AWS Management Console- Amazon Fraud Detector menyediakan antarmuka pengguna berbasis web, konsol Amazon Fraud Detector. Jika Anda mendaftarAkun AWS, Anda dapat mengakses konsol Amazon Fraud Detector. Untuk informasi selengkapnya, lihat Mengatur Amazon Fraud Detector.

AWS Command Line Interface(AWS CLI) - Menyediakan antarmuka yang dapat Anda gunakan untuk berinteraksi dengan kumpulan luasLayanan AWS, termasuk Amazon Fraud Detector, menggunakan perintah di shell baris perintah Anda. AWS CLIperintah untuk Amazon Fraud Detector menerapkan fungsionalitas yang setara dengan yang disediakan oleh konsol Amazon Fraud Detector.

AWSSDK - Menyediakan API khusus bahasa dan mengelola banyak detail koneksi, seperti perhitungan tanda tangan, penanganan coba ulang permintaan, dan penanganan kesalahan. Untuk informasi selengkapnya, buka AWS halaman Alat untuk membangun, gulir ke bawah ke bagian SDK, dan pilih tanda plus (+) untuk memperluas bagian.

AWS CloudFormation- Menyediakan template yang dapat Anda gunakan untuk menentukan sumber daya dan properti Amazon Fraud Detector Anda. Untuk informasi selengkapnya, lihat referensi jenis sumber daya Amazon Fraud Detector di Panduan AWS CloudFormation Pengguna.

Harga

Dengan Amazon Fraud Detector, Anda hanya membayar untuk apa yang Anda gunakan. Tidak ada biaya minimum atau pun komitment di muka Anda dikenakan biaya berdasarkan jam komputasi yang digunakan untuk melatih dan meng-host model Anda, jumlah penyimpanan yang Anda gunakan, dan jumlah prediksi penipuan yang Anda buat. Untuk informasi selengkapnya, lihat harga Amazon Fraud Detector.

Siapkan untuk Amazon Fraud Detector

Untuk menggunakan Amazon Fraud Detector, pertama-tama Anda memerlukan akun Amazon Web Services (AWS) dan kemudian Anda harus menyiapkan izin yang memberikan Akun AWS akses ke semua antarmuka. Kemudian ketika Anda mulai membuat sumber daya Amazon Fraud Detector, Anda perlu memberikan izin yang memungkinkan Amazon Fraud Detector mengakses akun Anda untuk melakukan tugas atas nama Anda dan mengakses sumber daya yang Anda miliki.

Selesaikan tugas-tugas berikut di bagian ini untuk menyiapkan penggunaan Amazon Fraud Detector:

- Daftar ke AWS.
- Siapkan izin yang memungkinkan Anda Akun AWS mengakses antarmuka Amazon Fraud Detector.
- Siapkan antarmuka yang ingin Anda gunakan untuk mengakses Amazon Fraud Detector.

Setelah Anda menyelesaikan langkah-langkah ini, lihat Memulai dengan Amazon Fraud Detector untuk melanjutkan memulai dengan Amazon Fraud Detector.

Daftar untuk AWS

Saat Anda mendaftar ke Amazon Web Services (AWS), Anda Akun AWS secara otomatis mendaftar untuk semua layananAWS, termasuk Amazon Fraud Detector. Anda hanya akan dikenakan biaya untuk layanan yang digunakan. Jika Anda sudah memiliki Akun AWS, lewati dan langsung ke tugas berikutnya.

Daftar Akun AWS

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

Untuk mendaftar Akun AWS

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

   Anda akan diminta untuk menerima panggilan telepon dan memasukkan kode verifikasi pada keypad telepon sebagai bagian dari prosedur pendaftaran.

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

AWS akan mengirimkan email konfirmasi kepada Anda setelah proses pendaftaran selesai. Anda dapat melihat aktivitas akun saat ini dan mengelola akun dengan mengunjungi https://aws.amazon.com/ dan memilih Akun Saya.

Membuat pengguna administratif

Setelah Anda mendaftarAkun AWS, amankanPengguna root akun AWS, aktifkanAWS IAM Identity Center, dan buat pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

Mengamankan Pengguna root akun AWS Anda

1. Masuk ke AWS Management Console sebagai pemilik akun dengan memilih Pengguna root dan memasukkan alamat email Akun AWS Anda. Pada halaman berikutnya, masukkan kata sandi Anda.

   Untuk bantuan masuk menggunakan pengguna root, lihat Masuk sebagai pengguna root dalam Panduan Pengguna AWS Sign-In.

2. Mengaktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

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

Membuat pengguna administratif

1. Aktifkan Pusat Identitas IAM.

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

2. Di Pusat Identitas IAM, berikan akses administratif ke pengguna administratif.

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

Masuk sebagai pengguna administratif

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

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

Siapkan izin untuk mengakses antarmuka Amazon Fraud Detector

Untuk menggunakan Amazon Fraud Detector, siapkan izin untuk mengakses konsol Amazon Fraud Detector dan operasi API.

Mengikuti praktik terbaik keamanan, buat pengguna AWS Identity and Access Management (IAM) dengan akses terbatas pada operasi Amazon Fraud Detector dan dengan izin yang diperlukan. Anda dapat menambahkan izin lain sesuai kebutuhan.

Kebijakan berikut memberikan izin yang diperlukan untuk menggunakan Amazon Fraud Detector:

- `AmazonFraudDetectorFullAccessPolicy`

  Memungkinkan Anda melakukan tindakan berikut:

  - Akses semua sumber daya Amazon Fraud Detector
  - Buat daftar dan jelaskan semua titik akhir model di SageMaker
  - Buat daftar semua peran IAM di akun
  - Daftar semua ember Amazon S3
  - Izinkan IAM Pass Role untuk meneruskan peran ke Amazon Fraud Detector
- `AmazonS3FullAccess`

  Memungkinkan akses penuh keAmazon Simple Storage Service. Ini diperlukan jika Anda perlu mengunggah kumpulan data pelatihan ke Amazon S3.

Berikut ini menjelaskan cara membuat pengguna IAM dan menetapkan izin yang diperlukan.

Untuk membuat pengguna dan menetapkan izin yang diperlukan

1. Masuk ke AWS Management Console dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Users (Pengguna) lalu pilih Add user(Tambahkan pengguna).
3. Untuk Nama pengguna, masukkan **`AmazonFraudDetectorUser`**.
4. Pilih kotak centang akses AWS Management Console, lalu konfigurasikan kata sandi pengguna.
5. (Opsional) Secara AWS default, pengguna baru harus membuat kata sandi baru saat pertama kali masuk. Anda dapat mengosongkan kotak centang di samping Pengguna harus membuat kata sandi baru saat masuk berikutnya agar pengguna baru dapat mengatur ulang kata sandi mereka setelah masuk.
6. Pilih Next: Permissions (Selanjutnya: Izin).
7. Pilih Create group (Buat grup).
8. Untuk nama Grup, masukkan**`AmazonFraudDetectorGroup`**.
9. Dalam daftar kebijakan, pilih kotak centang untuk AmazonFraudDetectorFullAccessPolicydan FullAccessAmazonS3. Pilih Create group (Buat grup).
10. Dalam daftar grup, pilih kotak centang untuk grup baru Anda. Pilih Segarkan jika Anda tidak melihat grup dalam daftar.
11. Pilih Next: Tags (Selanjutnya: Tanda).
12. (Opsional) Tambahkan metadata ke pengguna dengan cara melampirkan tanda sebagai pasangan nilai kunci. Untuk petunjuk tentang cara menggunakan tag di IAM, lihat Menandai Pengguna dan Peran IAM.
13. Pilih Berikutnya: Tinjau untuk melihat rincian Pengguna dan ringkasan Izin untuk pengguna baru. Saat Anda siap untuk melanjutkan, pilih Buat pengguna.

Siapkan antarmuka untuk mengakses Amazon Fraud Detector dengan

Anda dapat mengakses Amazon Fraud Detector menggunakan konsol Amazon Fraud DetectorAWS CLI, atau AWS SDK. Sebelum Anda dapat menggunakannya, pertama mengatur AWS CLI dan AWS SDK.

Akses konsol Amazon Fraud Detector

Anda dapat mengakses konsol Amazon Fraud Detector dan AWS layanan lainnya melaluiAWS Management Console. AndaAkun AWS, memberi Anda akses ke. AWS Management Console

Untuk mengakses konsol Amazon Fraud Detector,

1. Pergi ke https://console.aws.amazon.com/ dan masuk ke AndaAkun AWS.
2. Arahkan ke Amazon Fraud Detector.

Dengan konsol Amazon Fraud Detector, Anda dapat membuat dan mengelola model serta sumber daya deteksi penipuan seperti Detektor, Variabel, Peristiwa, Entitas, Label, dan Hasil. Anda dapat menghasilkan prediksi dan mengevaluasi kinerja dan prediksi model Anda.

Mengatur AWS CLI

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk berinteraksi dengan Amazon Fraud Detector dengan menjalankan perintah di shell baris perintah Anda. Dengan konfigurasi minimal, Anda dapat menggunakan perintah AWS CLI to run untuk fungsionalitas serupa dengan yang disediakan oleh konsol Amazon Fraud Detector dari prompt perintah di terminal Anda.

Untuk mengatur AWS CLI

Unduh dan konfigurasi AWS CLI. Untuk petunjuk, lihat topik berikut di Panduan AWS Command Line Interface Pengguna:

- Menyiapkan dengan Antarmuka Baris AWS Perintah
- Mengkonfigurasi Antarmuka Baris AWS Perintah

Untuk informasi tentang perintah Amazon Fraud Detector, lihat Perintah yang Tersedia

Siapkan AWS SDK

Anda dapat menggunakan AWS SDK untuk menulis kode untuk membuat dan mengelola sumber daya deteksi penipuan Anda dan untuk mendapatkan prediksi penipuan. AWSSDK mendukung Amazon Fraud Detector di JavaScriptdan Python (Boto3).

Untuk mengatur AWS SDK for Python (Boto3)

Anda dapat menggunakan AWS SDK for Python (Boto3) untuk membuat, mengkonfigurasi, dan mengelola AWS layanan. Untuk petunjuk tentang cara menginstal Boto, lihat AWSSDK for Python (Boto3). Pastikan Anda menggunakan Boto3 SDK versi 1.14.29 atau lebih tinggi.

Setelah Anda menginstalAWS SDK for Python (Boto3), jalankan contoh Python berikut untuk mengonfirmasi bahwa lingkungan Anda dikonfigurasi dengan benar. Jika dikonfigurasi dengan benar, responsnya berisi daftar detektor. Jika tidak ada detektor yang dibuat, daftarnya kosong.

```
import boto3
fraudDetector = boto3.client('frauddetector')

response = fraudDetector.get_detectors()
print(response)
```

Untuk mengatur AWS SDK untuk Java

Untuk petunjuk tentang cara menginstal dan memuatAWS SDK for JavaScript, lihat Menyiapkan SDK untuk JavaScript.

Memulai dengan Amazon Fraud Detector

Sebelum memulai, pastikan Anda telah membacaMendeteksi penipuan dengan Amazon Fraud Detector dan menyelesaikan langkah-langkahSiapkan untuk Amazon Fraud Detector.

Gunakan tutorial interaktif di bagian ini untuk mempelajari cara menggunakan Amazon Fraud Detector untuk membuat, melatih, dan menerapkan model deteksi penipuan. Untuk tutorial ini, Anda mengasumsikan peran analis penipuan menggunakan model pembelajaran mesin untuk memprediksi apakah pendaftaran akun baru adalah penipuan. Model harus dilatih menggunakan data dari pendaftaran akun. Amazon Fraud Detector memberikan contoh dataset pendaftaran akun untuk tutorial ini. Contoh dataset harus diunggah sebelum Anda memulai tutorial.

Anda dapat memulai dengan Amazon Fraud Detector menggunakan salah satu antarmuka berikut. Sebelum memulai dengan tutorial, pastikan Anda mengikuti instruksiMendapatkan dan meng-upload contoh dataset

- Tutorial: Mulai menggunakan konsol Amazon Fraud Detector
- Tutorial: Mulai menggunakanAWS SDK for Python (Boto3)

Mendapatkan dan meng-upload contoh dataset

Contoh dataset yang Anda gunakan dalam tutorial ini memberikan rincian pendaftaran akun online. Dataset berada dalam file teks yang menggunakan nilai dipisahkan koma (CSV) dalam format UTF-8. Baris pertama file set data CSV berisi header. Baris header diikuti oleh beberapa baris data. Masing-masing baris ini terdiri dari elemen data dari pendaftaran akun tunggal. Data diberi label demi kenyamanan Anda. Kolom dalam kumpulan data mengidentifikasi apakah pendaftaran akun itu curang.

Untuk mendapatkan dan mengunggah contoh dataset

1. Pergi ke Sampel.

   Ada dua file data yang memiliki data pendaftaran akun online - registration_data_20K_minimum.csv dan registration_data_20K_full.csv. File hanya`registration_data_20K_minimum` berisi dua variabel: ip_address dan email_address. File`registration_data_20K_full` berisi variabel lain. Variabel-variabel ini untuk setiap acara dan mereka termasuk billing_address, phone_number, dan user_agent. Kedua file data juga berisi dua bidang wajib:

- EVENT_TIMESTAMP - Mendefinisikan kapan peristiwa terjadi
- EVENT_LABEL - Mengklasifikasikan peristiwa sebagai penipuan atau sah

Anda dapat menggunakan salah satu dari dua file untuk tutorial ini. Unduh file data yang ingin Anda gunakan.

2. Buat bucketAmazon Simple Storage Service (Amazon S3).

   Pada langkah ini, Anda membuat penyimpanan eksternal untuk menyimpan set data. Penyimpanan eksternal ini adalah bucket Amazon S3. Untuk informasi lebih lanjut tentang Amazon S3, lihat Apa itu Amazon S3?

   a. Masuk ke AWS Management Console dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.

   b. Di Bucket, pilih Buat bucket.

   c. Untuk Nama bucket, masukkan nama bucket. Pastikan Anda mengikuti aturan penamaan bucket di konsol, dan berikan nama unik secara global. Sebaiknya gunakan nama yang menjelaskan tujuan bucket.

   d. Untuk Wilayah AWS, pilihWilayah AWS tempat Anda ingin membuat bucket. Wilayah yang Anda pilih harus mendukung Amazon Fraud Detector. Untuk mengurangi latensi, pilihWilayah AWS yang paling dekat dengan lokasi geografis Anda. Untuk daftar Wilayah yang mendukung Amazon Fraud Detector, lihat Tabel Wilayah di Panduan Infrastruktur Global.

   e. Biarkan pengaturan default untuk Kepemilikan Objek, pengaturan Bucket untuk Block Public Access, Bucket Versioning, dan Tag untuk tutorial ini.

   f. Untuk enkripsi Default, pilih Nonaktifkan untuk tutorial ini.

   g. Tinjau konfigurasi bucket, dan kemudian pilih Buat bucket.

3. Unggah file data contoh ke bucket Amazon S3.

   Sekarang setelah Anda memiliki bucket, unggah salah satu contoh file yang Anda unduh sebelumnya ke bucket Amazon S3 yang baru saja Anda buat.

   a. Di Bucket, nama bucket Anda dicantumkan. Pilih bucket Anda.

   b. Pilih Upload (Unggah).

   c. Di File dan folder, pilih Tambahkan file.

d. Pilih salah satu contoh file data yang Anda unduh di komputer Anda, lalu pilih Buka.

e. Biarkan pengaturan default untuk Tujuan, Izin, dan Properti.

f. Tinjau konfigurasi, lalu pilih Unggah.

g. Contoh file data diunggah ke bucket Amazon S3. Perhatikan lokasi bucket. Di Objects, pilih contoh file data yang baru saja Anda upload.

h. Dalam ikhtisar Objek, salin lokasi di bawah URI S3. Ini adalah lokasi Amazon S3 dari file data contoh. Anda menggunakannya nanti. Anda juga dapat menyalin Amazon Resource Name (ARN) dari bucket S3 dan menyimpannya.

Tutorial: Mulai menggunakan konsol Amazon Fraud Detector

Tutorial ini terdiri dari dua bagian. Bagian pertama menjelaskan cara membuat, melatih, dan menerapkan model deteksi penipuan. Bagian kedua mencakup cara menggunakan model untuk menghasilkan prediksi penipuan secara real time. Model dilatih menggunakan file data contoh yang Anda unggah ke bucket S3. Di akhir tutorial ini, Anda menyelesaikan tindakan berikut:

- Bangun dan latih model Amazon Fraud Detector
- Hasilkan prediksi penipuan waktu nyata

> **Important**
>
> Sebelum melanjutkan, pastikan bahwa Anda telah mengikuti instruksiMendapatkan dan meng-upload contoh dataset

Bagian A: Membangun, melatih, dan menerapkan model Amazon Fraud Detector

Di bagian A, Anda menentukan kasus penggunaan bisnis Anda, menentukan acara Anda, membuat model, melatih model, mengevaluasi kinerja model, dan menerapkan model.

Langkah 1: Pilih kasus penggunaan bisnis

- Pada langkah ini, Anda menggunakan penjelajah model data untuk mencocokkan kasus penggunaan bisnis Anda dengan jenis model deteksi penipuan yang didukung oleh Amazon Fraud Detector. Data models explorer adalah alat yang terintegrasi dengan konsol Amazon

Fraud Detector yang merekomendasikan jenis model yang akan digunakan untuk membuat dan melatih model deteksi penipuan untuk kasus penggunaan bisnis Anda. Penjelajah model data juga memberikan wawasan tentang elemen data wajib, direkomendasikan, dan opsional yang perlu Anda sertakan dalam kumpulan data Anda. Dataset akan digunakan untuk membuat dan melatih model deteksi penipuan Anda.

Untuk tujuan tutorial ini, kasus penggunaan bisnis Anda adalah pendaftaran akun baru. Setelah Anda menentukan kasus penggunaan bisnis Anda, penjelajah model data akan merekomendasikan jenis model untuk membuat model deteksi penipuan dan juga akan memberi Anda daftar elemen data yang Anda perlukan untuk membuat kumpulan data Anda. Karena Anda telah mengunggah kumpulan data sampel yang berisi data dari pendaftaran akun baru, Anda tidak perlu membuat kumpulan data baru.

a. Buka KonsolAWS Manajemen dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.

b. Di panel navigasi kiri, pilih Penjelajah model data.

c. Di halaman Penjelajah model data, di bawah Kasus penggunaan bisnis, pilih Penipuan akun baru.

d. Amazon Fraud Detector menampilkan jenis model yang direkomendasikan untuk digunakan untuk membuat model deteksi penipuan untuk kasus penggunaan bisnis yang dipilih. Jenis model mendefinisikan algoritme, pengayaan, dan transformasi yang akan digunakan Amazon Fraud Detector untuk melatih model deteksi penipuan Anda.

   Perhatikan tipe model yang direkomendasikan. Anda akan memerlukan ini nanti saat membuat model.

e. Panel wawasan model data memberikan wawasan tentang elemen data wajib dan direkomendasikan yang diperlukan untuk membuat dan melatih model deteksi penipuan.

   Lihatlah contoh dataset yang Anda download dan pastikan bahwa ia memiliki semua wajib dan beberapa elemen data yang direkomendasikan tercantum dalam tabel.

   Kemudian ketika Anda membuat model untuk kasus penggunaan bisnis spesifik Anda, Anda akan menggunakan wawasan yang disediakan untuk membuat kumpulan data Anda.

Langkah 2: Membuat jenis peristiwa

- Pada langkah ini, Anda menentukan aktivitas bisnis (acara) untuk mengevaluasi penipuan. Mendefinisikan acara melibatkan pengaturan variabel yang ada di kumpulan data, acara yang memulai entitas, dan label yang menggolkan acara. Untuk tutorial ini, Anda menentukan acara pendaftaran akun.

  a. Buka KonsolAWS Manajemen dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.

  b. Di panel navigasi kiri, pilih peristiwa.

  c. Di halaman Jenis peristiwa, pilih Buat.

  d. Di bawah Detail jenis peristiwa, masukkan`sample_registration` sebagai nama jenis acara dan, secara opsional, masukkan deskripsi acara.

  e. Untuk Entitas, pilih Buat entitas.

  f. Di halaman Buat entitas, masukkan`sample_customer` sebagai nama jenis entitas. Secara opsional, masukkan deskripsi tipe entitas.

  g. Pilih Buat entitas.

  h. Di bawah Variabel peristiwa, untuk Pilih cara menentukan variabel acara ini, pilih Pilih variabel dari kumpulan data pelatihan.

  i. Untuk peran IAM, pilih Buat peran IAM.

  j. Di halaman Buat peran IAM, masukkan nama bucket S3 tempat Anda mengunggah data contoh dan pilih Buat peran.

  k. Di Lokasi data, masukkan jalur ke data contoh Anda. Ini adalah`S3 URI` jalur yang Anda simpan setelah mengunggah data contoh. Jalannya mirip dengan ini:`S3://`*`your-bucket-name/example dataset filename`*`.csv`.

  l. Pilih Upload (Unggah).

     Amazon Fraud Detector mengekstrak header dari file data contoh Anda dan memetakannya dengan tipe variabel. Pemetaan ditampilkan di konsol.

  m. Di bawah Label - opsional, untuk Label, pilih Buat label baru.

  n. Dalam Buat halaman label, masukkan`fraud` sebagai nama. Label ini sesuai dengan nilai yang mewakili pendaftaran akun penipuan dalam kumpulan data contoh.

  o. Pilih Buat label.

p. Buat label kedua, lalu masukkan`legit` sebagai nama. Label ini sesuai dengan nilai yang mewakili pendaftaran akun yang sah dalam kumpulan data contoh.

q. Pilih Buat jenis acara.

Langkah 3: Membuat Model

1. Pada halaman Model, pilih Add model, lalu pilih Create model.
2. Untuk Langkah 1 - Tentukan detail model, masukkan`sample_fraud_detection_model` sebagai nama model. Secara opsional, tambahkan deskripsi model.
3. Untuk Tipe Model, pilih model Wawasan Penipuan Online.
4. Untuk jenis Event, pilih sample_registration. Ini adalah jenis peristiwa yang Anda buat di Langkah 1.
5. Dalam Data peristiwa historis,

   a. Di Sumber data peristiwa, pilih Data peristiwa yang disimpan di S3.

   b. Untuk peran IAM, pilih peran yang Anda buat di Langkah 1.

   c. Di Lokasi data pelatihan, masukkan jalur URI S3 ke file data contoh Anda.
6. Pilih Selanjutnya.

Langkah 4: Model kereta

1. Di input Model, biarkan semua kotak centang dicentang. Secara default, Amazon Fraud Detector menggunakan semua variabel dari kumpulan data peristiwa historis Anda sebagai input model.
2. Dalam Klasifikasi Label, untuk label Penipuan pilih penipuan karena label ini sesuai dengan nilai yang mewakili peristiwa penipuan dalam kumpulan data contoh. Untuk label yang sah, pilih legit karena label ini sesuai dengan nilai yang mewakili peristiwa yang sah dalam kumpulan data contoh.
3. Untuk perawatan peristiwa tak berlabel, simpan pilihan default Abaikan peristiwa yang tidak berlabel untuk kumpulan data contoh ini.
4. Pilih Selanjutnya.
5. Setelah meninjau, pilih Buat dan latih model. Amazon Fraud Detector membuat model dan mulai melatih versi baru model.

   Dalam versi Model kolom Status menunjukkan status pelatihan model. Pelatihan model yang menggunakan kumpulan data contoh membutuhkan waktu sekitar 45 menit untuk

menyelesaikannya. Status berubah menjadi Siap untuk diterapkan setelah pelatihan model selesai.

Langkah 5: Tinjau kinerja model

Langkah penting dalam menggunakan Amazon Fraud Detector adalah menilai keakuratan model Anda menggunakan skor model dan metrik kinerja. Setelah pelatihan model selesai, Amazon Fraud Detector memvalidasi kinerja model menggunakan 15% data Anda yang tidak digunakan untuk melatih model dan menghasilkan skor kinerja model dan metrik kinerja lainnya.

1. Untuk melihat kinerja model,

   a. Pada panel navigasi kiri konsol Fraud Detector pilih Model.

   b. Di halaman Model, pilih model yang baru saja Anda latih (sample_fraud_detection_model), lalu pilih 1.0. Ini adalah versi Amazon Fraud Detector yang dibuat dari model Anda.

2. Lihatlah skor keseluruhan kinerja Model dan semua metrik lain yang dihasilkan Amazon Fraud Detector untuk model ini.

   Untuk mempelajari lebih lanjut tentang skor kinerja model dan metrik kinerja di halaman ini, lihatSkor model danMetrik kinerja model.

   Anda dapat mengharapkan semua model Amazon Fraud Detector terlatih memiliki metrik kinerja deteksi penipuan dunia nyata yang mirip dengan metrik kinerja yang Anda lihat untuk model dalam tutorial ini.

Langkah 6: Deploy model

Setelah Anda meninjau metrik kinerja model terlatih Anda dan siap menggunakannya menghasilkan prediksi penipuan, Anda dapat menerapkan model.

1. Di panel navigasi kiri konsol Amazon Fraud Detector, pilih Model.
2. Di halaman Model, pilih sample_fraud_detection_model, lalu pilih versi model spesifik yang ingin Anda terapkan. Untuk tutorial ini, pilih 1.0.
3. Pada halaman versi Model, pilih Tindakan dan kemudian pilih Deploy versi model.
4. Dalam versi Model, Status menunjukkan status penyebaran. Status berubah menjadi Aktif setelah deployment selesai. Ini menunjukkan bahwa versi model diaktifkan dan tersedia untuk

menghasilkan prediksi penipuan. Lanjutkan dengan[Bagian B: Menghasilkan prediksi penipuan](#) untuk menyelesaikan langkah-langkah untuk menghasilkan prediksi penipuan.

Bagian B: Menghasilkan prediksi penipuan

Prediksi penipuan adalah evaluasi kecurangan untuk kegiatan bisnis (event). Amazon Fraud Detector menggunakan detektor untuk menghasilkan prediksi penipuan. Detektor berisi logika deteksi, seperti model dan aturan, untuk peristiwa tertentu yang ingin Anda evaluasi untuk penipuan. Logika deteksi menggunakan aturan untuk memberi tahu Amazon Fraud Detector cara menafsirkan data yang terkait dengan model. Dalam tutorial ini, Anda mengevaluasi peristiwa pendaftaran akun menggunakan set data contoh pendaftaran akun yang Anda unggah sebelumnya.

Di Bagian A, Anda membuat, melatih, dan menerapkan model Anda. Di Bagian B, Anda membuat detektor untuk jenis`sample_registration` peristiwa, menambahkan model yang diterapkan, membuat aturan dan urutan eksekusi aturan, lalu membuat dan mengaktifkan versi detektor yang Anda gunakan untuk menghasilkan prediksi penipuan.

Langkah 1: Bangun detektor

Membuat Detector

1. Pada panel navigasi kiri Fraud Detector Detektor.
2. Pilih Buat detektor.
3. Di halaman Tentukan detail detektor, masukkan`sample_detector` nama detektor. Secara opsional, masukkan deskripsi untuk detektor, seperti`my sample fraud detector`.
4. Untuk Jenis Acara, pilih sample_registration. Ini adalah acara yang Anda buat di Bagian A dari tutorial ini.
5. Pilih Selanjutnya.

Langkah 2: Menambahkan model

Jika Anda menyelesaikan Bagian A dari tutorial ini, maka Anda mungkin sudah memiliki model Amazon Fraud Detector yang tersedia untuk ditambahkan ke detektor Anda. Jika Anda belum membuat model, buka Bagian A dan selesaikan langkah-langkah untuk membuat, melatih, dan menerapkan model dan kemudian lanjutkan dengan Bagian B.

1. Dalam model Add - opsional, pilih Add Model.

2. Di halaman Tambahkan model, untuk model Pilih, pilih nama model Amazon Fraud Detector yang Anda gunakan sebelumnya. Untuk Pilih versi, pilih versi model model yang dikerahkan.
3. Pilih Tambahkan model.
4. Pilih Selanjutnya.

Langkah 3: Tambahkan aturan

Aturan adalah kondisi yang memberitahu Amazon Fraud Detector untuk menafkan skor kinerja model saat mengevaluasi prediksi penipuan. Untuk tutorial ini, Anda membuat tiga aturan:`high_fraud_risk`,`medium_fraud_risk`, dan`low_fraud_risk`.

1. Di halaman Tambahkan aturan, di bawah Tentukan aturan, masukkan`high_fraud_risk` nama aturan dan di bawah Deskripsi - opsional, masukkan**`This rule captures events with a high ML model score`** sebagai deskripsi untuk aturan.
2. Dalam Ekspresi, masukkan ekspresi aturan berikut menggunakan bahasa ekspresi aturan sederhana Amazon Fraud Detector:

   ```
   $sample_fraud_detection_model_insightscore > 900
   ```
3. Di Hasil, pilih Buat hasil baru. Hasilnya adalah hasil dari prediksi penipuan dan dikembalikan jika aturan cocok selama evaluasi.
4. Di Buat hasil baru, masukkan`verify_customer` sebagai nama hasil. Secara opsional, masukkan deskripsi.
5. Pilih Simpan hasil.
6. Pilih Tambahkan aturan untuk menjalankan pemeriksa validasi aturan dan menyimpan aturan. Setelah dibuat, Amazon Fraud Detector membuat aturan tersedia untuk digunakan di detektor Anda.
7. Pilih Tambahkan aturan lain, lalu pilih tab Buat aturan.
8. Ulangi proses ini dua kali lebih banyak untuk membuat`low_fraud_risk` aturan`medium_fraud_risk` dan aturan menggunakan rincian aturan berikut:

   - medium_fraud_risk

     Nama aturan:`medium_fraud_risk`

     Hasil:`review`

     Ekspresi:

```
$sample_fraud_detection_model_insightscore <= 900 and

$sample_fraud_detection_model_insightscore > 700
```

- low_fraud_risk

  Nama aturan:`low_fraud_risk`

  Hasil:`approve`

  Ekspresi:

  ```
  $sample_fraud_detection_model_insightscore <= 700
  ```

Nilai-nilai ini adalah contoh yang digunakan untuk tutorial ini. Saat Anda membuat aturan untuk detektor Anda sendiri, gunakan nilai yang sesuai untuk model dan kasus penggunaan Anda,

9. Setelah Anda membuat ketiga aturan, pilih Berikutnya.

   Untuk informasi lebih lanjut tentang membuat dan menulis aturan, lihatAturan danReferensi bahasa aturan.

Langkah 4: Konfigurasikan eksekusi aturan dan urutan aturan

Mode eksekusi aturan untuk aturan yang disertakan dalam detektor menentukan apakah semua aturan yang Anda tetapkan dievaluasi, atau jika evaluasi aturan berhenti pada aturan pertama yang cocok. Dan urutan aturan menentukan urutan yang Anda inginkan aturan untuk dijalankan.

Mode eksekusi aturan default adalah`FIRST_MATCHED`.

Pertama cocok

Mode eksekusi aturan pertama yang cocok mengembalikan hasil untuk aturan pencocokan pertama berdasarkan urutan aturan yang ditentukan. Jika Anda menentukan `FIRST_MATCHED`, Amazon Fraud Detector mengevaluasi aturan secara berurutan, pertama hingga terakhir, dengan berhenti pada aturan pertama yang cocok. Amazon Fraud Detector kemudian menyediakan hasil untuk aturan tunggal tersebut.

Urutan Anda menjalankan aturan dapat memengaruhi hasil prediksi penipuan yang dihasilkan. Setelah Anda membuat aturan, pesan ulang aturan untuk menjalankannya sesuai urutan yang diinginkan dengan mengikuti langkah-langkah berikut:

Jikahigh_fraud_risk aturan Anda belum berada di bagian atas daftar aturan Anda, pilih Pesanan, lalu pilih 1. Ini bergerakhigh_fraud_risk ke posisi pertama.

Ulangi proses ini sehinggamedium_fraud_risk aturan Anda berada di posisi kedua danlow_fraud_risk aturan Anda berada di posisi ketiga.

Semua cocok

Semua mode eksekusi aturan yang cocok mengembalikan hasil untuk semua aturan yang cocok, terlepas dari urutan aturan. Jika Anda menentukanALL_MATCHED, Amazon Fraud Detector mengevaluasi semua aturan dan mengembalikan hasil untuk semua aturan yang cocok.

PilihFIRST_MATCHED untuk tutorial ini dan kemudian pilih Berikutnya.

Langkah 5: Tinjau dan buat versi detektor

Versi detektor mendefinisikan model dan aturan spesifik yang digunakan untuk menghasilkan prediksi penipuan.

1. Di halaman Tinjau dan buat, tinjau detail detektor, model, dan aturan yang Anda konfigurasikan. Jika Anda perlu melakukan perubahan, pilih Edit di sebelah bagian yang sesuai.
2. Pilih Buat detektor. Setelah dibuat, versi pertama detektor Anda muncul di tabel versi Detector denganDraft status.

   Anda menggunakan versi Draft untuk menguji Detector Anda.

Langkah 6: Uji dan aktifkan versi detektor

Di konsol Amazon Fraud Detector, Anda dapat menguji logika detektor Anda menggunakan data tiruan dengan fitur Run test. Untuk tutorial ini, Anda dapat menggunakan data pendaftaran akun dari contoh dataset.

1. Gulir ke Jalankan pengujian di bagian bawah halaman detail versi Detektor.
2. Untuk metadata peristiwa, masukkan stempel waktu saat peristiwa terjadi dan masukkan pengenal unik untuk entitas yang melakukan acara tersebut. Untuk tutorial ini, pilih tanggal dari pemilih tanggal untuk timestamp, dan masukkan “1234” untuk ID Entitas.
3. Untuk variabel Event, masukkan nilai variabel yang ingin Anda uji. Untuk tutorial ini, Anda hanya perluip_address danemail_address bidang. Ini karena mereka adalah input yang digunakan

untuk melatih model Amazon Fraud Detector Anda. Anda dapat menggunakan nilai contoh berikut. Ini mengasumsikan bahwa Anda menggunakan nama variabel yang disarankan:

- alamat:`205.251.233.178`
- alamat email:`johndoe@exampledomain.com`

4. Memilih Menjalankan pengujian.
5. Amazon Fraud Detector mengembalikan hasil prediksi penipuan berdasarkan mode eksekusi aturan. Jika mode eksekusi aturan`FIRST_MATCHED`, hasil yang dikembalikan sesuai dengan aturan pertama yang cocok. Aturan pertama adalah aturan dengan prioritas tertinggi. Ini cocok jika dievaluasi sebagai benar. Jika mode eksekusi aturan`ALL_MATCHED`, hasil yang dikembalikan sesuai dengan semua aturan yang cocok. Itu berarti bahwa mereka semua dievaluasi untuk menjadi kenyataan. Amazon Fraud Detector juga mengembalikan skor model untuk model apa pun yang ditambahkan ke detektor Anda.

   Anda dapat mengubah input dan menjalankan beberapa tes untuk melihat hasil yang berbeda. Anda dapat menggunakan nilai ip_address dan email_address dari kumpulan data contoh Anda untuk pengujian dan memeriksa apakah hasilnya seperti yang diharapkan.
6. Ketika Anda puas dengan cara kerja detektor, promosikan dari`Draft` ke`Active`. Melakukan hal itu membuat detektor tersedia untuk digunakan dalam deteksi penipuan real-time.

   Pada halaman Detail versi Detektor, pilih Tindakan, Publikasikan, Publikasikan versi. Ini mengubah status detektor dari Draft ke Aktif.

   Pada titik ini, model Anda dan logika detektor terkait siap untuk mengevaluasi aktivitas online untuk penipuan secara real time menggunakan Amazon Fraud Detector`GetEventPrediction` API. Anda juga dapat mengevaluasi peristiwa secara offline menggunakan file input CSV dan`CreateBatchPredictionJob` API. Untuk informasi lebih lanjut tentang prediksi penipuan, lihatPrediksi penipuan

Dengan menyelesaikan tutorial ini, Anda melakukan hal berikut:

- Mengunggah set data peristiwa contoh ke Amazon S3.
- Membuat dan melatih model deteksi penipuan Amazon Fraud Detector menggunakan set data contoh.
- Melihat skor kinerja model dan metrik kinerja lain yang dihasilkan Amazon Fraud Detector.
- Menerapkan model deteksi penipuan.

- Membuat detektor dan menambahkan model yang digunakan.
- Aturan tambahan, perintah eksekusi aturan, dan hasil ke detektor.
- Menguji detektor dengan memberikan input yang berbeda dan memeriksa apakah aturan dan perintah eksekusi aturan bekerja seperti yang diharapkan.
- Mengaktifkan detektor dengan menerbitkannya.

Tutorial: Mulai menggunakanAWS SDK for Python (Boto3)

Tutorial ini menjelaskan cara membangun dan melatih model Amazon Fraud Detector dan kemudian menggunakan model ini untuk menghasilkan prediksi penipuan real-time menggunakanAWS SDK for Python (Boto3). Model ini dilatih menggunakan file data contoh pendaftaran akun yang Anda unggah ke bucket Amazon S3.

Pada akhir tutorial ini, Anda menyelesaikan tindakan berikut:

- Bangun dan latih model Amazon Fraud Detector
- Buat prediksi penipuan waktu nyata

Prasyarat

Berikut ini adalah langkah-langkah prasyarat untuk tutorial ini.

- SelesaiSiapkan untuk Amazon Fraud Detector.

  Jika sudahSiapkan AWS SDK, pastikan Anda menggunakan Boto3 SDK versi 1.14.29 atau yang lebih tinggi.
- Ikuti instruksi untukMendapatkan dan meng-upload contoh dataset file yang diperlukan untuk tutorial ini.

Memulai

Langkah 1: Siapkan dan verifikasi lingkungan Python Anda

Boto adalah SDK untuk Python Amazon Web Services (AWS). Anda dapat menggunakannya untuk membuat, mengkonfigurasi, dan mengelolaLayanan AWS. Untuk petunjuk tentang cara menginstal Boto3, lihat AWS SDK for Python (Boto3).

Setelah Anda menginstalAWS SDK for Python (Boto3), jalankan perintah contoh Python berikut untuk mengkonfirmasi bahwa lingkungan Anda dikonfigurasi dengan benar. Jika lingkungan Anda dikonfigurasi dengan benar, respons berisi daftar detektor. Jika tidak ada detektor yang dibuat, daftar tersebut kosong.

```
import boto3
fraudDetector = boto3.client('frauddetector')

response = fraudDetector.get_detectors()
print(response)
```

Langkah 2: Buat variabel, jenis entitas, dan label

Pada langkah ini, Anda membuat sumber daya yang digunakan untuk menentukan model, peristiwa, dan aturan.

Buat variabel

Variabel adalah elemen data dari dataset Anda yang ingin Anda gunakan untuk membuat jenis acara, model, dan aturan.

Pada contoh berikut, CreateVariableAPI digunakan untuk membuat dua variabel. Variabel-variabelnya adalah`email_address` dan`ip_address`. Menetapkan mereka untuk jenis variabel yang sesuai:`EMAIL_ADDRESS` dan`IP_ADDRESS`. Variabel-variabel ini adalah bagian dari contoh dataset yang Anda upload. Saat Anda menentukan jenis variabel, Amazon Fraud Detector menafsirkan variabel selama pelatihan model dan saat mendapatkan prediksi. Hanya variabel dengan tipe variabel terkait yang dapat digunakan untuk pelatihan model.

```
import boto3
fraudDetector = boto3.client('frauddetector')

 #Create variable email_address
fraudDetector.create_variable(
    name = 'email_address',
    variableType = 'EMAIL_ADDRESS',
    dataSource = 'EVENT',
    dataType = 'STRING',
    defaultValue = '<unknown>'
)

#Create variable ip_address
```

```
fraudDetector.create_variable(
    name = 'ip_address',
    variableType = 'IP_ADDRESS',
    dataSource = 'EVENT',
    dataType = 'STRING',
    defaultValue = '<unknown>'
)
```

Buat tipe entitas

Entitas mewakili siapa yang melakukan peristiwa dan jenis entitas mengklasifikasikan entitas. Klasifikasi contoh termasuk pelanggan, pedagang, atau akun.

Pada contoh berikut, PutEntityTypeAPI digunakan untuk membuat jenis`sample_customer` entitas.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_entity_type(
   name = 'sample_customer',
   description = 'sample customer entity type'
)
```

Buat label

Label mengklasifikasikan peristiwa sebagai penipuan atau sah dan digunakan untuk melatih model deteksi penipuan. Model belajar untuk mengklasifikasikan peristiwa menggunakan nilai-nilai label ini.

Pada contoh berikut, API Putlabel digunakan untuk membuat dua label,`fraud` dan`legit`.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_label(
    name = 'fraud',
    description = 'label for fraud events'
)

fraudDetector.put_label(
```

```
    name = 'legit',
    description = 'label for legitimate events'
)
```

Langkah 3: Buat jenis acara

Dengan Amazon Fraud Detector, Anda membuat model yang mengevaluasi risiko dan menghasilkan prediksi penipuan untuk peristiwa individual. Jenis acara mendefinisikan struktur peristiwa individu.

Pada contoh berikut, PutEventTypeAPI digunakan untuk membuat jenis acara`sample_registration`. Anda menentukan jenis peristiwa dengan menentukan variabel (`email_address`,`ip_address`), jenis entitas (`sample_customer`), dan label (`fraud`,`legit`) yang Anda buat pada langkah sebelumnya.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_event_type (
     name = 'sample_registration',
     eventVariables = ['ip_address', 'email_address'],
     labels = ['legit', 'fraud'],
     entityTypes = ['sample_customer'])
```

Langkah 4: Membuat, melatih, dan menerapkan model

Amazon Fraud Detector melatih model untuk belajar mendeteksi penipuan untuk jenis peristiwa tertentu. Pada langkah sebelumnya, Anda membuat jenis peristiwa. Pada langkah ini, Anda membuat dan melatih model untuk jenis acara. Model bertindak sebagai wadah untuk versi model Anda. Setiap kali Anda melatih model, versi baru dibuat.

Gunakan kode contoh berikut untuk membuat dan melatih model Wawasan Penipuan Online. Model ini disebut`sample_fraud_detection_model`. Ini untuk jenis peristiwa`sample_registration` menggunakan set data contoh pendaftaran akun yang Anda unggah ke Amazon S3.

Untuk informasi selengkapnya tentang berbagai jenis model yang didukung Amazon Fraud Detector, lihatPilih jenis model.

Buat model

Pada contoh berikut, CreateModelAPI digunakan untuk membuat model.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_model (
       modelId = 'sample_fraud_detection_model',
       eventTypeName = 'sample_registration',
       modelType = 'ONLINE_FRAUD_INSIGHTS')
```

Melatih model

Pada contoh berikut, CreateModelVersionAPI digunakan untuk melatih model. Tentukan`'EXTERNAL_EVENTS'` untuk`trainingDataSource` dan lokasi Amazon S3 tempat Anda menyimpan kumpulan data contoh dan bucket Amazon S3`externalEventsDetail`. RoleArn Untuk`trainingDataSchema` parameter, tentukan bagaimana Amazon Fraud Detector menafsirkan data contoh. Lebih khusus lagi, tentukan variabel mana yang akan disertakan dan bagaimana mengklasifikasikan label acara.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_model_version (
         modelId = 'sample_fraud_detection_model',
         modelType = 'ONLINE_FRAUD_INSIGHTS',
         trainingDataSource = 'EXTERNAL_EVENTS',
         trainingDataSchema = {
            'modelVariables' : ['ip_address', 'email_address'],
            'labelSchema' : {
               'labelMapper' : {
                   'FRAUD' : ['fraud'],
                   'LEGIT' : ['legit']
        }
    }
},
         externalEventsDetail = {
              'dataLocation' : 's3://your-S3-bucket-name/your-example-data-
filename.csv',
              'dataAccessRoleArn' : 'role_arn'
}
)
```

Anda dapat melatih model Anda beberapa kali. Setiap kali Anda melatih model, versi baru dibuat. Setelah pelatihan model selesai, status versi model diperbaruiTRAINING_COMPLETE. Anda dapat meninjau skor kinerja model dan metrik kinerja model lainnya.

Tinjau kinerja model

Langkah penting dalam menggunakan Amazon Fraud Detector adalah menilai keakuratan model Anda menggunakan skor model dan metrik kinerja. Setelah pelatihan model selesai, Amazon Fraud Detector memvalidasi kinerja model menggunakan 15% data Anda yang tidak digunakan untuk melatih model. Ini menghasilkan skor kinerja model dan metrik kinerja lainnya.

Gunakan DescribeModelVersionsAPI untuk meninjau performa model. Lihat skor keseluruhan kinerja Model dan semua metrik lain yang dihasilkan oleh Amazon Fraud Detector untuk model ini.

Untuk mempelajari lebih lanjut tentang skor kinerja model dan metrik kinerja, lihatSkor model danMetrik kinerja model.

Anda dapat mengharapkan semua model Amazon Fraud Detector terlatih memiliki metrik kinerja deteksi penipuan dunia nyata, yang mirip dengan metrik dalam tutorial ini.

Menerapkan model

Setelah Anda meninjau metrik kinerja model terlatih Anda, terapkan model dan sediakan untuk Amazon Fraud Detector untuk menghasilkan prediksi penipuan. Untuk menerapkan model terlatih, gunakan UpdateModelVersionStatusAPI. Pada contoh berikut, digunakan untuk memperbarui status versi model ke ACTIVE.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_model_version_status (
     modelId = 'sample_fraud_detection_model',
     modelType = 'ONLINE_FRAUD_INSIGHTS',
     modelVersionNumber = '1.00',
     status = 'ACTIVE'
)
```

Langkah 5: Buat versi detektor, hasil, aturan, dan detektor

Detektor berisi logika deteksi, seperti model dan aturan. Logika ini untuk acara tertentu yang ingin Anda evaluasi untuk penipuan. Aturan adalah suatu kondisi yang ditentukan untuk memberitahu

Amazon Fraud Detector mengenai cara menafsirkan nilai variabel selama prediksi. Dan hasilnya adalah hasil dari prediksi penipuan. Detektor dapat memiliki beberapa versi dengan setiap versi memiliki status DRAFT, AKTIF, atau TIDAK AKTIF. Versi detektor harus memiliki setidaknya satu aturan yang terkait dengannya.

Gunakan contoh kode berikut untuk membuat detektor, aturan, hasil, dan untuk mempublikasikan detektor.

Buat detektor

Pada contoh berikut, PutDetectorAPI digunakan untuk membuat`sample_detector` detektor untuk jenis`sample_registration` peristiwa.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_detector (
     detectorId = 'sample_detector',
     eventTypeName = 'sample_registration'
)
```

Ciptakan hasil

Hasil dibuat untuk setiap kemungkinan hasil prediksi penipuan. Pada contoh berikut, PutOutcomeAPI digunakan untuk membuat tiga hasil -`verify_customer`,`review`, dan`approve`. Hasil ini kemudian ditugaskan untuk aturan.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_outcome(
     name = 'verify_customer',
     description = 'this outcome initiates a verification workflow'
    )

fraudDetector.put_outcome(
     name = 'review',
     description = 'this outcome sidelines event for review'
    )

fraudDetector.put_outcome(
```

```
     name = 'approve',
     description = 'this outcome approves the event'
)
```

Buat aturan

Aturan terdiri dari satu atau lebih variabel dari dataset Anda, ekspresi logika, dan satu atau lebih hasil.

Pada contoh berikut, CreateRuleAPI digunakan untuk membuat tiga aturan yang berbeda:`high_risk`,`medium_risk`, dan`low_risk`. Buat ekspresi aturan untuk membandingkan`sample_fraud_detection_model_insightscore` nilai skor kinerja model dengan berbagai ambang batas. Hal ini untuk menentukan tingkat risiko untuk suatu peristiwa dan menetapkan hasil yang didefinisikan pada langkah sebelumnya.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_rule(
     ruleId = 'high_fraud_risk',
     detectorId = 'sample_detector',
     expression = '$sample_fraud_detection_model_insightscore > 900',
     language = 'DETECTORPL',
     outcomes = ['verify_customer']
     )

fraudDetector.create_rule(
     ruleId = 'medium_fraud_risk',
     detectorId = 'sample_detector',
     expression = '$sample_fraud_detection_model_insightscore <= 900 and
 $sample_fraud_detection_model_insightscore > 700',
     language = 'DETECTORPL',
     outcomes = ['review']
     )

fraudDetector.create_rule(
     ruleId = 'low_fraud_risk',
     detectorId = 'sample_detector',
     expression = '$sample_fraud_detection_model_insightscore <= 700',
     language = 'DETECTORPL',
     outcomes = ['approve']
```

```
      )
```

Buat versi detektor

Versi detektor mendefinisikan model dan aturan yang digunakan untuk mendapatkan prediksi penipuan.

Pada contoh berikut, CreateDetectorVersionAPI digunakan untuk membuat versi detektor. Hal ini dilakukan dengan menyediakan rincian versi model, aturan, dan modus eksekusi aturan FIRST_MATCH. Mode eksekusi aturan menentukan urutan untuk mengevaluasi aturan. Modus eksekusi aturan FIRST_MATCHED menentukan bahwa aturan dievaluasi secara berurutan, pertama hingga terakhir, dengan berhenti pada aturan pertama yang cocok.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_detector_version(
      detectorId = 'sample_detector',
      rules = [{
          'detectorId' : 'sample_detector',
          'ruleId' : 'high_fraud_risk',
          'ruleVersion' : '1'
},
{
          'detectorId' : 'sample_detector',
          'ruleId' : 'medium_fraud_risk',
          'ruleVersion' : '1'
},
{
          'detectorId' : 'sample_detector',
          'ruleId' : 'low_fraud_risk',
          'ruleVersion' : '1'
}
],
      modelVersions = [{
          'modelId' : 'sample_fraud_detection_model',
          'modelType': 'ONLINE_FRAUD_INSIGHTS',
          'modelVersionNumber' : '1.00'
}      ],
      ruleExecutionMode = 'FIRST_MATCHED'
)
```

Langkah 6: Buat prediksi penipuan

Langkah terakhir dari tutorial ini menggunakan detektor yang`sample_detector` dibuat pada langkah sebelumnya untuk menghasilkan prediksi penipuan untuk jenis`sample_registration` peristiwa secara real time. Detector mengevaluasi data contoh yang diunggah ke Amazon S3. Tanggapannya mencakup skor kinerja model serta hasil apa pun yang terkait dengan aturan yang cocok.

Pada contoh berikut, GetEventPredictionAPI digunakan untuk menyediakan data dari pendaftaran akun tunggal dengan setiap permintaan. Untuk tutorial ini, ambil data (email_address dan ip_address) dari file data contoh pendaftaran akun. Setiap baris (baris) setelah baris header atas mewakili data dari acara pendaftaran akun tunggal.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.get_event_prediction(
      detectorId = 'sample_detector',
      eventId = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
      eventTypeName = 'sample_registration',
      eventTimestamp = '2020-07-13T23:18:21Z',
      entities = [{'entityType':'sample_customer', 'entityId':'12345'}],
 eventVariables = {
      'email_address': 'johndoe@exampledomain.com',
      'ip_address': '1.2.3.4'
}
)
```

Setelah Anda menyelesaikan tutorial ini, Anda melakukan hal berikut:

- Unggah kumpulan data peristiwa contoh ke Amazon S3.
- Variabel, entitas, dan label yang dibuat yang digunakan untuk membuat dan melatih model.
- Membuat dan melatih model menggunakan set data contoh.
- Melihat skor kinerja model dan metrik kinerja lain yang dihasilkan Amazon Fraud Detector.
- Menerapkan model deteksi penipuan.

- Membuat detektor dan menambahkan model yang digunakan.
- Aturan tambahan, perintah eksekusi aturan, dan hasil ke detektor.
- Dibuat versi detektor.
- Menguji detektor dengan memberikan input yang berbeda dan memeriksa apakah aturan dan perintah eksekusi aturan bekerja seperti yang diharapkan.

(Opsional) Jelajahi API Amazon Fraud Detector dengan Notebook Jupyter (iPython)

Untuk contoh lainnya tentang cara menggunakan Amazon Fraud Detector API, lihat aws-fraud-detector-samples GitHub repositori. Topik yang dicakup notebook mencakup model bangunan dan detektor menggunakan Amazon Fraud Detector API dan membuat permintaan prediksi penipuan batch menggunakan`GetEventPrediction` API.

Langkah selanjutnya

Sekarang setelah Anda membuat model dan detektor, Anda dapat menyelam lebih dalam dan mulai membuat model dan detektor serta menghasilkan prediksi penipuan.

Bagian berikut dalam Panduan Pengguna Amazon Fraud Detector menjelaskan bagaimana bisnis atau organisasi Anda dapat menggunakan Amazon Fraud Detector untuk mendeteksi penipuan.

- Siapkan dan buat kumpulan data acara Anda untuk melatih model Anda.
- Membuat jenis peristiwa
- Buat model
- Buat detektor
- Dapatkan prediksi penipuan
- Mengelola sumber daya Amazon Fraud Detector Anda (khususnya, Variabel, Entitas, Hasil, dan Label)
- Konfigurasikan Amazon Fraud Detector
- Memantau Amazon Fraud Detector dan log panggilan Amazon Fraud Detector API
- Pemecah masalah dengan Amazon Fraud Detector

Set data data peristiwa

Dataset peristiwa adalah data penipuan historis untuk perusahaan Anda. Anda memberikan data ini ke Amazon Fraud Detector untuk membuat model deteksi penipuan.

Amazon Fraud Detector menggunakan model machine learning untuk menghasilkan prediksi penipuan. Setiap model dilatih menggunakan tipe model. Jenis model menentukan algoritma dan transformasi yang digunakan untuk melatih model. Pelatihan model adalah proses menggunakan kumpulan data yang Anda berikan untuk membuat model yang dapat memprediksi kejadian penipuan. Untuk informasi selengkapnya, lihat Cara kerja Amazon Fraud Detector

Dataset yang digunakan untuk membuat model deteksi penipuan memberikan rincian peristiwa. Acara adalah aktivitas bisnis yang dievaluasi akan risiko penipuan. Misalnya, pendaftaran akun bisa menjadi acara. Data yang terkait dengan acara pendaftaran akun dapat berupa kumpulan data acara. Amazon Fraud Detector menggunakan kumpulan data ini untuk mengevaluasi penipuan pendaftaran akun.

Sebelum Anda memberikan dataset Anda ke Amazon Fraud Detector untuk membuat model, pastikan untuk menentukan tujuan Anda untuk membuat model. Anda juga perlu menentukan bagaimana Anda ingin menggunakan model dan menentukan metrik Anda untuk mengevaluasi apakah model berkinerja berdasarkan kebutuhan spesifik Anda.

Misalnya, tujuan Anda untuk membuat model deteksi penipuan yang mengevaluasi penipuan pendaftaran akun adalah sebagai berikut:

- Untuk menyetujui pendaftaran yang sah secara otomatis.
- Untuk menangkap pendaftaran penipuan untuk penyelidikan nanti.

Setelah Anda menentukan tujuan Anda, langkah selanjutnya adalah memutuskan bagaimana Anda ingin menggunakan model. Beberapa contoh untuk menggunakan model deteksi penipuan untuk mengevaluasi penipuan pendaftaran adalah sebagai berikut:

- Untuk deteksi penipuan real-time untuk setiap pendaftaran akun.
- Untuk evaluasi offline semua pendaftaran akun setiap jam.

Beberapa contoh metrik yang dapat digunakan untuk mengukur kinerja model meliputi:

- Melakukan secara konsisten lebih baik daripada baseline saat ini dalam produksi.

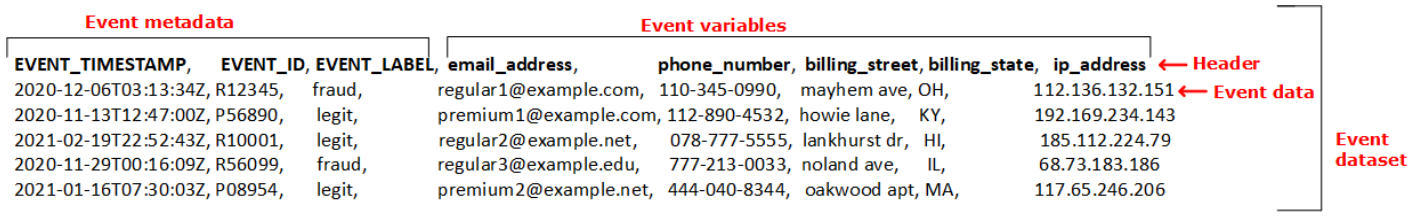
- Menangkap pendaftaran penipuan X% dengan tingkat positif palsu Y%.
- Menerima hingga 5% dari pendaftaran yang disetujui secara otomatis yang bersifat penipuan.

Struktur data data peristiwa

Amazon Fraud Detector mengharuskan Anda menyediakan kumpulan data peristiwa dalam file teks menggunakan nilai dipisahkan koma (CSV) dalam format UTF-8. Baris pertama file set data CSV Anda harus berisi header file. Header file terdiri dari metadata peristiwa dan variabel acara yang menggambarkan setiap elemen data yang terkait dengan acara tersebut. Header diikuti oleh data acara. Setiap baris terdiri dari elemen data dari satu peristiwa.

- Data peristiwa - memberikan informasi tentang peristiwa tersebut. Misalnya, EVENT_TIMESTAMP adalah metadata peristiwa yang menentukan peristiwa waktu terjadi. Bergantung pada kasus penggunaan bisnis Anda dan jenis model yang digunakan untuk membuat dan melatih model deteksi penipuan Anda, Amazon Fraud Detector mengharuskan Anda untuk menyediakan metadata peristiwa tertentu. Saat menentukan metadata peristiwa di header file CSV Anda, gunakan nama metadata peristiwa yang sama seperti yang ditentukan oleh Amazon Fraud Detector dan gunakan huruf besar saja.
- Variabel peristiwa - mewakili elemen data yang spesifik untuk acara Anda yang ingin Anda gunakan untuk membuat dan melatih model deteksi penipuan Anda. Bergantung pada kasus penggunaan bisnis Anda dan jenis model yang digunakan untuk membuat dan melatih model deteksi penipuan, Amazon Fraud Detector mungkin memerlukan atau menyarankan Anda untuk menyediakan variabel peristiwa tertentu. Anda juga dapat menyediakan variabel acara lain dari acara Anda yang ingin Anda sertakan dalam melatih model. Beberapa contoh variabel acara untuk acara pendaftaran online dapat berupa alamat email, alamat ip, dan nomor telepon. Saat menentukan nama variabel acara di header file CSV Anda, gunakan nama variabel pilihan Anda dan gunakan huruf kecil saja.
- Data acara - mewakili data yang dikumpulkan dari acara aktual. Dalam file CSV Anda, setiap baris yang mengikuti header file terdiri dari elemen data dari satu peristiwa. Misalnya, dalam file data acara pendaftaran online, setiap baris berisi data dari satu pendaftaran. Setiap elemen data dalam baris harus cocok dengann metadata peristiwa yang sesuai atau variabel acara.

Berikut ini adalah contoh file CSV yang berisi data dari peristiwa pendaftaran akun. Baris header berisi metadata peristiwa dalam variabel huruf besar dan acara dalam huruf kecil diikuti oleh data acara. Setiap baris dalam dataset berisi elemen data yang terkait dengan pendaftaran akun tunggal dengan setiap elemen data yang sesuai dengan header.



Dapatkan persyaratan dataset acara menggunakan data model explorer

Jenis model yang Anda pilih untuk membuat model menentukan persyaratan untuk kumpulan data Anda. Amazon Fraud Detector menggunakan kumpulan data yang Anda berikan untuk membuat dan melatih model deteksi penipuan Anda. Sebelum Amazon Fraud Detector mulai membuat model Anda, Amazon Fraud Detector akan memeriksa apakah dataset memenuhi ukuran, format, dan persyaratan lainnya. Jika kumpulan data tidak memenuhi persyaratan, pembuatan dan pelatihan model gagal. Anda dapat menggunakan penjelajah model data untuk mengidentifikasi jenis model yang akan digunakan untuk kasus penggunaan bisnis Anda dan untuk mendapatkan wawasan tentang persyaratan set data untuk jenis model yang diidentifikasi.

Model data

Penjelajah model data adalah alat di konsol Amazon Fraud Detector yang menyelaraskan kasus penggunaan bisnis Anda dengan jenis model yang didukung oleh Amazon Fraud Detector. Penjelajah model data juga memberikan wawasan tentang elemen data yang diperlukan oleh Amazon Fraud Detector untuk membuat model deteksi penipuan Anda. Sebelum Anda mulai menyiapkan kumpulan data peristiwa, gunakan penjelajah model data untuk mengetahui jenis model yang direkomendasikan Amazon Fraud Detector untuk penggunaan bisnis Anda dan juga untuk melihat daftar elemen data wajib, direkomendasikan, dan opsional yang Anda perlukan untuk membuat kumpulan data Anda.

Untuk menggunakan data model explorer,

1. Lanjutkan konsolAWS manajemen dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi di sebelah kiri, pilih Data models explorer.
3. Di halaman Penjelajah model data, dalam kasus penggunaan Bisnis, pilih kasus penggunaan bisnis yang ingin Anda evaluasi untuk risiko penipuan.

4. Amazon Fraud Detector menampilkan jenis model yang direkomendasikan yang cocok dengan kasus penggunaan bisnis Anda. Jenis model mendefinisikan algoritme, pengayaan, dan transformasi yang akan digunakan Amazon Fraud Detector untuk melatih model deteksi penipuan Anda.

   Perhatikan tipe model yang direkomendasikan. Anda akan membutuhkan ini nanti saat Anda membuat model Anda.

   > **Note**
   >
   > Jika Anda tidak menemukan kasus penggunaan bisnis Anda, gunakan tautan hubungi kami dalam deskripsi untuk memberikan detail kasus penggunaan bisnis Anda kepada kami. Kami akan merekomendasikan jenis model yang akan digunakan untuk membuat model deteksi penipuan untuk kasus penggunaan bisnis Anda.

5. Panel wawasan model data memberikan wawasan tentang elemen data wajib, direkomendasikan, dan opsional yang diperlukan untuk membuat dan melatih model deteksi penipuan untuk kasus penggunaan bisnis Anda. Gunakan informasi di panel wawasan untuk mengumpulkan data acara Anda dan membuat kumpulan data Anda.

Mengumpulkan data peristiwa

Mengumpulkan data acara Anda adalah langkah penting dalam membuat model Anda. Ini karena kinerja model Anda dalam memprediksi penipuan tergantung pada kualitas kumpulan data Anda. Saat Anda mulai mengumpulkan data acara Anda, ingatlah daftar elemen data yang disediakan penjelajah model Data bagi Anda untuk membuat kumpulan data Anda. Anda perlu mengumpulkan semua data wajib (metadata peristiwa) dan memutuskan elemen data yang direkomendasikan dan opsional (variabel peristiwa) untuk disertakan berdasarkan tujuan Anda untuk membuat model. Penting juga untuk menentukan format setiap variabel peristiwa yang ingin Anda sertakan dan ukuran total kumpulan data Anda.

Kualitas set data acara

Untuk mengumpulkan set data berkualitas tinggi untuk model Anda, kami merekomendasikan hal berikut:

- Kumpulkan data matang- Menggunakan data terbaru membantu mengidentifikasi pola penipuan terbaru. Namun, untuk mendeteksi kasus penggunaan penipuan, izinkan data menjadi matang.

Masa jatuh tempo tergantung pada bisnis Anda, dan dapat berlangsung dari dua minggu hingga tiga bulan. Misalnya, jika acara Anda termasuk transaksi kartu kredit, maka jatuh tempo data mungkin ditentukan oleh periode tagihan balik kartu kredit atau waktu yang diambil oleh penyidik untuk membuat penentuan.

Pastikan bahwa dataset yang digunakan untuk melatih model memiliki waktu yang cukup untuk matang sesuai bisnis Anda.

- Pastikan distribusi data tidak melayang secara signifikan- sampel proses pelatihan model Amazon Fraud Detector dan mempartisi kumpulan data Anda berdasarkan EVENT_TIMESTAMP. Misalnya, jika kumpulan data Anda terdiri dari peristiwa penipuan yang diambil dari 6 bulan terakhir, tetapi hanya bulan terakhir peristiwa yang sah yang disertakan, distribusi data dianggap hanyut dan tidak stabil. Dataset yang tidak stabil dapat menyebabkan bias dalam evaluasi kinerja model. Jika Anda menemukan distribusi data melayang secara signifikan, pertimbangkan untuk menyeimbangkan kumpulan data Anda dengan mengumpulkan data yang mirip dengan distribusi data saat ini.
- Pastikan dataset mewakili kasus penggunaan di mana model diimplementasikan/diuji- Jika tidak, perkiraan kinerja bisa bias. Mari kita katakan bahwa Anda menggunakan model untuk secara otomatis menolak semua pelamar in-door, tetapi model Anda dilatih dengan dataset yang memiliki data/label historis yang sebelumnya disetujui. Kemudian, evaluasi model Anda mungkin tidak akurat karena evaluasi didasarkan pada kumpulan data yang tidak memiliki representasi dari pelamar yang ditolak.

Format data peristiwa

Amazon Fraud Detector mengubah sebagian besar data Anda ke format yang diperlukan sebagai bagian dari proses pelatihan modelnya. Namun, ada beberapa format standar yang dapat Anda gunakan dengan mudah untuk menyediakan data Anda yang dapat membantu menghindari masalah nanti saat Amazon Fraud Detector memvalidasi kumpulan data Anda. Tabel berikut memberikan panduan tentang format untuk menyediakan metadata peristiwa yang direkomendasikan.

> **Note**
>
> Saat Anda membuat file CSV, pastikan untuk memasukkan nama metadata peristiwa seperti yang tercantum di bawah ini, dalam huruf besar.

Nama metadata	Format	Diperlukan
EVENT_ID	Jika disediakan, itu harus memenuhi persyaratan berikut: • Hal ini unik untuk acara itu. • Ini mewakili informasi yang berarti bagi bisnis Anda. • Ini mengikuti pola ekspresi reguler (misalnya,`^[0-9a-z_-]+$.`) • Selain persyaratan di atas, sebaiknya Anda tidak menambahkan stempel waktu ke EVENT_ID. Melakukan hal itu dapat menyebabkan masalah saat Anda memperbarui acara. Ini karena Anda harus memberikan EVENT_ID yang sama persis jika Anda melakukan ini.	Tergantung pada jenis model
EVENT_TIMESTAMP	• Ini harus ditentukan dalam salah satu format berikut: • %YYY-%mm-%ddT%hh: %mm: %ssZ (standar ISO 8601 dalam UTC hanya tanpa milidetik) Contoh: 2019-11-30T 13:01:01 Z • %yyyy/%mm/%dd %hh: %mm: %ss (AM/PM)	Ya

Nama metadata	Format	Diperlukan
	Contoh: 2019/11/30 13:01:01 PM, atau 2019/11/30 13:01:01 • %mm/%dd/%yyyy %hh:%mm: %ss Contoh: 30/11/2019 13:01:01 WIB, 30/11/2019 13:01:01 • %mm/%dd/%yy %hh:%mm: %ss Contoh: 11/30/19 13:01:01 PM, 11/30/19 13:01:01 • Amazon Fraud Detector membuat asumsi berikut saat mengurai format tanggal/stempel waktu untuk cap waktu peristiwa: • Jika Anda menggunakan standar ISO 8601, itu harus sama persis dengan spesifikasi sebelumnya • Jika Anda menggunakan salah satu format lain, ada fleksibilitas tambahan: • Selama berbulan-bulan dan berhari-hari, Anda dapat memberikan satu atau dua digit. Misalnya, 1/12/2019	

Nama metadata	Format	Diperlukan
	adalah tanggal yang valid. • Anda tidak perlu menyertakan hh:mm:ss jika Anda tidak memilikinya (yaitu, Anda cukup memberikan tanggal). Anda juga dapat memberikan subset hanya jam dan menit (misalnya, hh:mm). Hanya menyediakan jam tidak didukung. Milidetik juga tidak didukung. • Jika Anda memberikan label AM/PM, jam 12 jam diasumsikan. Jika tidak ada informasi AM/PM, jam 24 jam diasumsikan. • Anda dapat menggunakan “/” atau “-” sebagai pembatas untuk elemen tanggal. “:” diasumsikan untuk elemen timestamp.	

Nama metadata	Format	Diperlukan
ENTITY_ID	• Itu harus mengikuti pola ekspresi reguler:`^[0-9A-Za-z_.@+-]+$` . • Jika id entitas tidak tersedia pada saat evaluasi, tentukan id entitas sebagai tidak diketahui.	Tergantung pada jenis model
ENTITY_TYPE	Anda dapat menggunakan string apa pun	Tergantung pada jenis model
EVENT_LABEL	Anda dapat menggunakan label apa pun, seperti “penipuan”, “legit”, “1", atau “0".	Diperlukan jika LABEL_TIMESTAMP disertakan
LABEL_TIMESTAMP	Ini harus mengikuti format stempel waktu.	Diperlukan jika EVENT_LABEL disertakan

Untuk informasi tentang variabel acara, lihat Variabel.

> **Important**
>
> Jika Anda membuat model Account Takeover Insights (ATI), lihatMempersiapkan data detail tentang menyiapkan dan memilih data.

Nilai nol atau hilang

Variabel EVENT_TIMESTAMP dan EVENT_LABEL tidak boleh mengandung nilai null atau hilang. Anda dapat memiliki nilai nol atau hilang untuk variabel lain. Namun, kami menyarankan Anda hanya menggunakan sejumlah kecil untuk variabel tersebut. Jika Amazon Fraud Detector menentukan bahwa ada terlalu banyak nilai nol atau hilang untuk variabel peristiwa, maka secara otomatis akan menghilangkan variabel dari model Anda.

Variabel minimum

Saat Anda membuat model, kumpulan data harus menyertakan setidaknya dua variabel peristiwa selain metadata peristiwa yang diperlukan. Kedua variabel acara harus lulus pemeriksaan validasi.

Ukuran dataset acara

Diperlukan

Kumpulan data Anda harus memenuhi persyaratan dasar berikut untuk pelatihan model yang berhasil.

- Data dari setidaknya 100 peristiwa.
- Dataset harus menyertakan setidaknya 50 peristiwa (baris) yang diklasifikasikan sebagai penipuan.

Disarankan

Kami menyarankan agar kumpulan data Anda menyertakan yang berikut ini untuk pelatihan model yang sukses dan kinerja model yang baik.

- Sertakan minimal tiga minggu data historis, tetapi paling baik enam bulan data.
- Sertakan minimal 10K total data peristiwa.
- Sertakan setidaknya 400 peristiwa (baris) yang diklasifikasikan sebagai penipuan dan 400 peristiwa (baris) diklasifikasikan sebagai sah.
- Sertakan lebih dari 100 entitas unik, jika tipe model Anda memerlukan ENTITY_ID.

Validasi data

Sebelum Amazon Fraud Detector mulai membuat model Anda, Amazon Fraud Detector akan memeriksa apakah variabel yang disertakan dalam kumpulan data untuk melatih model memenuhi ukuran, format, dan persyaratan lainnya. Jika dataset tidak lulus validasi, model tidak dibuat. Anda harus terlebih dahulu memperbaiki variabel yang tidak lulus validasi sebelum Anda membuat model. Amazon Fraud Detector memberi Anda profil Data yang dapat Anda gunakan untuk membantu Anda mengidentifikasi dan memperbaiki masalah dengan kumpulan data Anda sebelum Anda mulai melatih model

Profiler data

Amazon Fraud Detector menyediakan alat sumber terbuka untuk membuat profil dan menyiapkan data Anda untuk pelatihan model. Profiler data otomatis ini membantu Anda menghindari kesalahan

persiapan data umum dan mengidentifikasi potensi masalah seperti jenis variabel yang salah dipetakan yang akan berdampak negatif pada kinerja model. Profiler menghasilkan laporan intuitif dan komprehensif dari kumpulan data Anda, termasuk statistik variabel, distribusi label, analisis kategoris dan numerik, dan korelasi variabel dan label. Ini memberikan panduan tentang jenis variabel serta opsi untuk mengubah kumpulan data menjadi format yang diperlukan Amazon Fraud Detector.

Menggunakan data profiler

Profiler data otomatis dibuat denganAWS CloudFormation tumpukan, yang dapat Anda luncurkan dengan mudah dengan beberapa klik. Semua kode tersedia di Github. Untuk informasi tentang cara menggunakan profiler data, ikuti petunjuk arah di blog kami Melatih model lebih cepat dengan profiler data otomatis untuk Amazon Fraud Detector

Kesalahan dataset peristiwa umum

Berikut ini adalah beberapa masalah umum yang dihadapi Amazon Fraud Detector saat memvalidasi kumpulan data peristiwa. Setelah Anda menjalankan profiler data, gunakan daftar ini untuk memeriksa kesalahan set data Anda sebelum membuat model Anda.

- File CSV tidak dalam format UTF-8.
- Jumlah peristiwa dalam dataset kurang dari 100.
- Jumlah kejadian yang diidentifikasi sebagai penipuan atau sah kurang dari 50.
- Jumlah entitas unik yang terkait dengan peristiwa penipuan kurang dari 100.
- Lebih dari 0,1% nilai dalam EVENT_TIMESTAMP berisi null atau nilai selain format tanggal/stempel waktu yang didukung.
- Lebih dari 1% dari nilai-nilai dalam EVENT_LABEL berisi nulls atau nilai-nilai selain yang didefinisikan dalam jenis peristiwa.
- Kurang dari dua variabel tersedia untuk pelatihan model.

Penyimpanan data data

Setelah set data, Anda menyimpan set data secara internal menggunakan Amazon Fraud Detector atau secara eksternal dengan Amazon Simple Storage Service (Amazon S3). Kami menyarankan Anda memilih tempat menyimpan kumpulan data berdasarkan model yang Anda gunakan untuk menghasilkan prediksi penipuan. Untuk informasi selengkapnya tentang jenis model, lihat Memilih

jenis model. Untuk informasi selengkapnya tentang menyimpan dataset Anda, lihatPenyimpanan data peristiwa.

Tipe peristiwa

Dengan Amazon Fraud Detector, Anda menghasilkan prediksi penipuan untuk acara. Jenis peristiwa mendefinisikan struktur untuk peristiwa individual yang dikirim ke Amazon Fraud Detector. Setelah ditentukan, Anda dapat membuat model dan detektor yang mengevaluasi risiko untuk jenis peristiwa tertentu.

Struktur suatu peristiwa meliputi:

- Entity Type: Mengklasifikasikan siapa yang melakukan acara tersebut. Selama prediksi, tentukan jenis entitas dan Id entitas untuk menentukan siapa yang melakukan acara tersebut.
- Variabel: Mendefinisikan variabel apa yang dapat dikirim sebagai bagian dari acara. Variabel digunakan oleh model dan aturan untuk mengevaluasi risiko penipuan. Setelah ditambahkan, variabel tidak dapat dihapus dari jenis acara.
- Label: Mengklasifikasikan acara sebagai penipuan atau sah. Digunakan selama pelatihan model. Setelah ditambahkan, label tidak dapat dihapus dari jenis acara.

Membuat jenis acara

Sebelum Anda membuat model deteksi penipuan, Anda harus terlebih dahulu membuat jenis peristiwa. Membuat jenis acara melibatkan mendefinisikan aktivitas bisnis Anda (acara) untuk mengevaluasi penipuan. Mendefinisikan peristiwa melibatkan mengidentifikasi variabel peristiwa dalam kumpulan data Anda untuk disertakan untuk evaluasi penipuan, menentukan entitas yang memulai acara, dan label yang mengklasifikasikan acara.

Prasyarat untuk membuat jenis acara

Sebelum Anda mulai membuat jenis acara Anda, pastikan bahwa Anda telah menyelesaikan hal berikut:

- Menggunakan Model data alat ini untuk mendapatkan wawasan tentang elemen data yang diperlukan oleh Amazon Fraud Detector untuk membuat model deteksi penipuan Anda.
- Menggunakan wawasan yang Anda dapatkan dari Data Models Explorer untuk membuat kumpulan data peristiwa dan telah mengunggah set data Anda ke bucket Amazon S3.
- DibuatVariabel,Entitas, dan Label Anda ingin Amazon Fraud Detector digunakan untuk membuat model deteksi penipuan untuk acara ini. Pastikan bahwa variabel, jenis entitas, dan label yang Anda buat disertakan dalam kumpulan data peristiwa Anda.

Anda dapat membuat jenis peristiwa di konsol Amazon Fraud Detector, menggunakan API, menggunakanAWS CLI, atau menggunakan AWS SDK.

Membuat jenis peristiwa di konsol Amazon Fraud Detector

Untuk membuat jenis acara,

1. Buka AWSManagement Console dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Acara.
3. Di halaman Jenis peristiwa, pilih Buat.
4. Di bawah Rincian jenis acara,,
   a. Dalam Nama, masukkan nama acara Anda.
   b. Dalam Deskripsi, secara opsional, masukkan deskripsi.
   c. Di Entitas, pilih jenis entitas yang Anda buat untuk acara Anda.
5. Di bawah variabel Event,
   - Dalam Pilih bagaimana mendefinisikan variabel acara ini,
     - Jika Anda telah membuat variabel acara untuk acara ini, pilih Pilih variabel dari daftar variabel Anda dan dalam Variabel, pilih variabel yang Anda buat untuk acara ini.
     - Jika Anda belum membuat variabel untuk acara ini, pilih Pilih variabel dari kumpulan data pelatihan,
       - Dalam peran IAM pilih Peran IAM yang Anda inginkan Amazon Fraud Detector untuk mengakses bucket Amazon S3 yang berisi kumpulan data Anda
       - Di Lokasi data masukkan jalur ke lokasi set data Anda. Gunakan `S3 URI` jalur yang mirip dengan ini:`S3://your-bucket-name/example dataset filename.csv`.
       - Pilih Upload (Unggah).
       - Di bawah Variabel, semua nama variabel peristiwa yang telah diambil Amazon Fraud Detector dari file set data Anda ditampilkan.

         Jika Anda ingin variabel disertakan untuk mendeteksi penipuan, dalam jenis Variabel, pilih jenis variabel. Pilih Hapus untuk menghapus variabel agar tidak disertakan untuk deteksi penipuan. Ulangi langkah ini untuk setiap variabel dalam daftar.
6. Di bawah Label (opsional), di Label, pilih label yang Anda buat untuk acara ini. Pastikan untuk memilih satu label masing-masing untuk acara penipuan dan sah.

7. Jika Anda ingin menyiapkan pemrosesan hilir otomatis untuk acara ini, di bawah Orkestrasi peristiwa dengan Amazon EventBridge - opsional, aktifkan Aktifkan orkestrasi peristiwa dengan Amazon. EventBridge Untuk informasi lebih lanjut tentang orkestrasi acara, lihat. Orkestrasi acara

   > **Note**
   >
   > Anda juga dapat mengaktifkan orkestrasi acara nanti setelah membuat jenis acara Anda.

8. Pilih Buat jenis acara.

Buat jenis acara menggunakan AWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan sampel untuk `PutEventType` API. Contoh mengasumsikan Anda telah membuat variabel `ip_address` dan`email_address`, label dan`fraud`, `legit` dan jenis entitas. `sample_customer` Untuk informasi tentang cara membuat sumber daya ini, lihatSumber daya.

> **Note**
>
> Anda harus terlebih dahulu membuat variabel, jenis entitas, dan label sebelum menambahkannya ke jenis acara.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_event_type (
name = 'sample_registration',
eventVariables = ['ip_address', 'email_address'],
labels = ['legit', 'fraud'],
entityTypes = ['sample_customer'])
```

Menghapus jenis peristiwa atau peristiwa

Saat Anda menghapus suatu peristiwa, Amazon Fraud Detector menghapus peristiwa tersebut secara permanen dan data yang terkait dengan peristiwa tersebut tidak lagi disimpan di Amazon Fraud Detector.

Untuk menghapus peristiwa yang telah dievaluasi Amazon Fraud Detector melalui API **`GetEventPrediction`**

1. Masuk ke AWS Management Console dan buka konsol Amazon Fraud Detector di https://console.aws.amazon.com/frauddetector.
2. Di panel navigasi kiri konsol, pilih Cari prediksi sebelumnya.
3. Pilih acara yang ingin Anda hapus.
4. Pilih Tindakan, lalu pilih Hapus acara.
5. Masukkan**`delete`**, lalu pilih Hapus acara.

> **Note**
>
> Ini menghapus semua data yang terkait dengan ID Peristiwa tersebut, termasuk data peristiwa apa pun yang dikirim ke `SendEvent` operasi dan data prediksi apa pun yang dihasilkan melalui operasi. `GetEventPrediction`

Untuk menghapus peristiwa yang disimpan di Amazon Fraud Detector tetapi belum dievaluasi (yaitu, itu disimpan melalui `SendEvent` operasi), Anda harus membuat `DeleteEvent` permintaan dan menentukan ID Peristiwa dan ID Jenis Peristiwa. Jika Anda ingin menghapus peristiwa dan riwayat prediksi apa pun yang terkait dengan peristiwa, tetapkan nilai `deleteAuditHistory` parameter ke “true”. Dengan `deleteAuditHistory` parameter diatur ke “true”, data acara tersedia melalui pencarian hingga 30 detik setelah operasi hapus selesai.

Untuk menghapus semua peristiwa yang terkait dengan jenis peristiwa

1. Di panel navigasi kiri konsol, pilih Jenis acara
2. Pilih jenis acara yang Anda ingin semua acara dihapus.
3. Arahkan ke tab Acara tersimpan dan pilih Hapus acara yang disimpan

Tergantung pada jumlah peristiwa yang disimpan untuk jenis acara, mungkin perlu beberapa waktu untuk menghapus semua peristiwa yang disimpan. Misalnya, kumpulan data 1 GB (sekitar 1-2 juta peristiwa untuk rata-rata pelanggan) membutuhkan waktu sekitar 2 jam untuk dihapus. Selama waktu ini, acara baru yang Anda kirim ke Amazon Fraud Detector jenis peristiwa ini tidak disimpan, tetapi Anda dapat terus menghasilkan prediksi penipuan melalui operasi. `GetEventPrediction`

Untuk menghapus jenis peristiwa

Anda tidak dapat menghapus jenis peristiwa yang digunakan dalam detektor atau model, atau telah terkait peristiwa yang disimpan. Sebelum menghapus jenis acara, Anda harus menghapus semua peristiwa yang terkait dengan jenis acara tersebut.

Saat Anda menghapus jenis peristiwa, Amazon Fraud Detector menghapus jenis peristiwa tersebut secara permanen dan data tidak lagi disimpan di Amazon Fraud Detector.

1. Di panel navigasi kiri konsol Amazon Fraud Detector, pilih Sumber Daya, lalu pilih Peristiwa.
2. Pilih jenis acara yang ingin Anda hapus.
3. Pilih Tindakan, lalu pilih Hapus jenis peristiwa.
4. Masukkan nama jenis acara, lalu pilih Hapus jenis acara.

Penyimpanan data peristiwa

Setelah Anda mengumpulkan set data, Anda menyimpan set data Anda secara internal menggunakan Amazon Fraud Detector atau secara eksternal dengan Amazon Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Storage Service (Amazon S3) Simple Kami menyarankan Anda memilih tempat menyimpan kumpulan data berdasarkan model yang Anda gunakan untuk menghasilkan prediksi penipuan. Berikut ini adalah rincian rinci dari dua opsi penyimpanan ini.

- Penyimpanan internal- Kumpulan data Anda disimpan dengan Amazon Fraud Detector. Semua data peristiwa yang terkait dengan suatu peristiwa disimpan bersama-sama. Anda dapat mengunggah kumpulan data peristiwa yang disimpan dengan Amazon Fraud Detector kapan saja. Anda dapat melakukan streaming peristiwa satu per satu ke Amazon Fraud Detector API, atau mengimpor kumpulan data besar (hingga 1 GB) menggunakan fitur impor batch. Saat Anda melatih model menggunakan kumpulan data yang disimpan dengan Amazon Fraud Detector, Anda dapat menentukan rentang waktu untuk membatasi ukuran set data Anda.
- Penyimpanan eksternal- Kumpulan data Anda disimpan di sumber data eksternal selain Amazon Fraud Detector. Saat ini, Amazon S3 Fraud Detector Simple Simple SSimple Simple Simple Simple Simple Simple Simple Simple S Jika model Anda berada di file yang diunggah ke Amazon S3, file tersebut tidak boleh melebihi 5GB data yang tidak terkompresi. Jika lebih dari itu, pastikan untuk mempersingkat rentang waktu kumpulan data Anda.

Tabel berikut memberikan detail tentang jenis model dan sumber data yang mendukung.

Tipe model	Sumber data pelatihan yang kompatibel
Wawasan Fraud Online	Penyimpanan eksternal, Penyimpanan internal
Wawasan Fraud Transaksi	Penyimpanan internal
Wawasan Pengambilalihan Akun	Penyimpanan internal

Untuk informasi tentang menyimpan kumpulan data Anda secara eksternal dengan Amazon Simple Storage Service, lihatMenyimpan data peristiwa Anda secara eksternal dengan Amazon S3 .
Untuk informasi tentang menyimpan dataset Anda secara internal dengan Amazon Fraud Detector lihatSimpan data acara Anda secara internal dengan Amazon Fraud Detector.

Menyimpan data peristiwa Anda secara eksternal dengan Amazon S3

Jika Anda melatih model Wawasan Penipuan Online, Anda dapat memilih untuk menyimpan data acara Anda secara eksternal dengan Amazon S3. Untuk menyimpan data peristiwa di Amazon S3, Anda harus terlebih dahulu membuat file teks dalam format CSV, menambahkan data peristiwa, dan kemudian mengunggah file CSV ke bucket Amazon S3.

> **Note**
>
> Jenis model Wawasan Penipuan Transaksi dan Wawasan Pengambilalihan Akun tidak mendukung kumpulan data yang disimpan secara eksternal dengan Amazon S3

Buat file CSV

Amazon Fraud Detector mengharuskan baris pertama file CSV Anda berisi header kolom. Header kolom dalam file CSV Anda harus memetakan ke variabel yang didefinisikan dalam jenis peristiwa. Untuk contoh dataset, lihat[Mendapatkan dan meng-upload contoh dataset]

Model Wawasan Penipuan Online memerlukan kumpulan data pelatihan yang memiliki setidaknya 2 variabel dan hingga 100 variabel. Selain variabel peristiwa, kumpulan data pelatihan harus berisi header berikut:

- EVENT_TIMESTAMP - Mendefinisikan kapan peristiwa terjadi
- EVENT_LABEL - Mengklasifikasikan peristiwa sebagai penipuan atau sah. Nilai-nilai dalam kolom harus sesuai dengan nilai-nilai yang didefinisikan dalam jenis acara.

Contoh data CSV berikut mewakili peristiwa pendaftaran historis dari pedagang online:

```
EVENT_TIMESTAMP,EVENT_LABEL,ip_address,email_address
4/10/2019 11:05,fraud,209.146.137.48,fake_burtonlinda@example.net
12/20/2018 20:04,legit,203.0.112.189,fake_davidbutler@example.org
3/14/2019 10:56,legit,169.255.33.54,fake_shelby76@example.net
1/3/2019 8:38,legit,192.119.44.26,fake_curtis40@example.com
9/25/2019 3:12,legit,192.169.85.29,fake_rmiranda@example.org
```

> **Note**
>
> File data CSV dapat berisi tanda kutip ganda dan koma sebagai bagian dari data Anda.

Versi yang disederhanakan dari jenis acara yang sesuai diwakili di bawah ini. Variabel acara sesuai dengan header dalam file CSV dan nilai-nilai`EVENT_LABEL` sesuai dengan nilai-nilai dalam daftar label.

```
(
name = 'sample_registration',
eventVariables = ['ip_address', 'email_address'],
labels = ['legit', 'fraud'],
entityTypes = ['sample_customer']
)
```

Format Timestamp Peristiwa

Pastikan stempel waktu acara Anda dalam format yang diperlukan. Sebagai bagian dari proses pembuatan model, jenis model Wawasan Penipuan Online memerintahkan data Anda berdasarkan stempel waktu peristiwa, dan membagi data Anda untuk tujuan pelatihan dan pengujian. Untuk mendapatkan perkiraan kinerja yang adil, model pertama melatih pada set data pelatihan, dan kemudian menguji model ini pada set data pengujian.

Amazon Fraud Detector mendukung format tanggal/stempel waktu berikut untuk nilai`EVENT_TIMESTAMP` selama pelatihan model:

- %YYY-%mm-%ddT%hh: %mm: %ssZ (standar ISO 8601 dalam UTC hanya tanpa milidetik)

  Contoh: 2019-11-30T 13:01:01 Z
- %yyyy/%mm/%dd %hh: %mm: %ss (AM/PM)

  Contoh: 2019/11/30 13:01:01 PM, atau 2019/11/30 13:01:01
- %mm/%dd/%yyyy %hh: %mm: %ss

  Contoh: 30/11/2019 13:01:01 WIB, 30/11/2019 13:01:01
- %mm/%dd/%yy %hh: %mm: %ss

  Contoh: 11/30/19 13:01:01 PM, 11/30/19 13:01:01

Amazon Fraud Detector membuat asumsi berikut saat mengurai format tanggal/stempel waktu untuk cap waktu peristiwa:

- Jika Anda menggunakan standar ISO 8601, itu harus sama persis dengan spesifikasi sebelumnya
- Jika Anda menggunakan salah satu format lain, ada fleksibilitas tambahan:
  - Selama berbulan-bulan dan berhari-hari, Anda dapat memberikan satu atau dua digit. Misalnya, 1/12/2019 adalah tanggal yang valid.
  - Anda tidak perlu menyertakan hh:mm:ss jika Anda tidak memilikinya (iya nih, Anda hanya dapat memberikan tanggal). Anda juga dapat memberikan subset hanya jam dan menit (misalnya, hh:mm). Hanya menyediakan jam tidak mendukung. Milidetik juga tidak didukung.
  - Jika Anda memberikan label AM/PM, jam 12 jam diasumsikan. Jika tidak ada informasi AM/PM, jam 24 jam diasumsikan.
  - Anda dapat menggunakan “/” atau “-” sebagai pembatas untuk elemen tanggal. “:” diasumsikan untuk elemen timestamp.

Sampling dataset Anda sepanjang waktu

Kami menyarankan Anda memberikan contoh penipuan dan sampel yang sah dari rentang waktu yang sama. Misalnya, jika Anda memberikan peristiwa penipuan dari 6 bulan terakhir, Anda juga harus menyediakan acara yang sah yang merata rentang periode waktu yang sama. Jika kumpulan data Anda berisi distribusi penipuan dan peristiwa yang sah yang sangat tidak merata, Anda mungkin menerima kesalahan berikut: “Distribusi penipuan di sepanjang waktu tidak dapat diterima secara fluktuasi. Tidak dapat membagi dataset dengan benar.” Biasanya, perbaikan termudah untuk kesalahan ini adalah memastikan bahwa peristiwa penipuan dan peristiwa yang sah disampel secara merata di jangka waktu yang sama. Anda juga mungkin perlu menghapus data jika Anda mengalami lonjakan besar dalam penipuan dalam jangka waktu singkat.

Jika Anda tidak dapat menghasilkan data yang cukup untuk membuat kumpulan data yang didistribusikan secara merata, salah satu pendekatannya adalah mengacak EVENT_TIMESTAMP peristiwa Anda sehingga didistribusikan secara merata. Namun, ini sering mengakibatkan metrik kinerja menjadi tidak realistis karena Amazon Fraud Detector menggunakan EVENT_TIMESTAMP untuk mengevaluasi model pada subset peristiwa yang sesuai dalam kumpulan data Anda.

Null dan nilai yang hilang

Amazon Fraud Detector menangani nilai nol dan nilai yang hilang. Namun, persentase nulls untuk variabel harus dibatasi. Kolom EVENT_TIMESTAMP dan EVENT_LABEL tidak boleh mengandung nilai yang hilang.

Validasi file

Amazon Fraud Detector akan gagal melatih model jika salah satu dari persyaratan berikut dipicu:

- Jika CSV tidak dapat diurai
- Jika tipe data untuk kolom salah

Unggah data peristiwa Anda ke bucket Amazon S3

Setelah Anda membuat file CSV dengan data peristiwa Anda, unggah file ke bucket Amazon S3 Anda.

Untuk mengunggah ke bucket Amazon S3

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Pilih Buat bucket.

   Buat bucket membuka wizard.
3. Di Nama bucket, masukkan nama yang sesuai dengan DNS untuk bucket Anda.

   Nama kelompok harus:

   - Unik di seluruh Amazon S3.
   - Panjangnya antara 3 dan 63 karakter.
   - Tidak mengandung karakter huruf besar.
   - Mulai dengan huruf atau angka kecil.

   Setelah membuat bucket, Anda tidak dapat mengubah namanya. Untuk informasi tentang penamaan, lihat Aturan penamaan bucket dalam Panduan Pengguna Amazon Simple Simple Simple Simple Simple Simple Simple Simple S

> **Important**
>
> Hindari memasukkan informasi sensitif, seperti nomor akun, dalam nama kelompok. Nama bucket terlihat dalam URL yang menunjuk objek dalam bucket.

4. Di Wilayah, pilihAWS Wilayah tempat Anda ingin bucket berada. Anda harus memilih Wilayah yang sama di mana Anda menggunakan Amazon Fraud Detector, US East (N. Virginia), US East (Ohio), US West (Oregon), Europe (Irlandia), Asia Pacific (Singapore) atau Asia Pacific (Sydney).
5. Pada Pengaturan bucket untuk Blokir Akses Publik, pilih pengaturan Blokir Akses Publik yang ingin Anda terapkan ke bucket.

   Kami menyarankan agar Anda membiarkan semua pengaturan diaktifkan. Untuk informasi selengkapnya tentang memblokir akses publik, lihat Memblokir akses publik ke penyimpanan Amazon S3 Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple S
6. Pilih Create bucket (Buat bucket).
7. Unggah file data pelatihan ke bucket Amazon S3 Anda. Perhatikan jalur lokasi Amazon S3 untuk file latihan Anda (misalnya, s3://bucketname/object.csv).

Simpan data acara Anda secara internal dengan Amazon Fraud Detector

Anda dapat memilih untuk menyimpan data peristiwa di Amazon Fraud Detector dan menggunakan data yang disimpan nanti untuk melatih model Anda. Dengan menyimpan data peristiwa di Amazon Fraud Detector, Anda dapat melatih model yang menggunakan variabel komputasi otomatis untuk meningkatkan kinerja, menyederhanakan pelatihan ulang model, dan memperbarui label penipuan untuk menutup loop umpan balik machine learning. Peristiwa disimpan di tingkat sumber daya Jenis Peristiwa, sehingga semua peristiwa dari jenis peristiwa yang sama disimpan bersama dalam satu set data jenis peristiwa. Sebagai bagian dari menentukan jenis peristiwa, Anda dapat secara opsional menentukan apakah akan menyimpan peristiwa untuk jenis peristiwa tersebut dengan mengubah setelan Penyerapan Peristiwa di konsol Amazon Fraud Detector.

Anda dapat menyimpan peristiwa tunggal atau mengimpor sejumlah besar dataset peristiwa di Amazon Fraud Detector. Peristiwa tunggal dapat dialirkan menggunakan GetEventPredictionAPI atau SendEventAPI. Set data besar dapat dengan cepat dan mudah diimpor ke Amazon Fraud

Detector menggunakan fitur impor batch di konsol Amazon Fraud Detector atau menggunakan CreateBatchImportJobAPI.

Anda dapat menggunakan konsol Amazon Fraud Detector kapan saja untuk memeriksa jumlah peristiwa yang telah disimpan untuk setiap jenis peristiwa.

Siapkan data acara untuk penyimpanan

Data peristiwa yang disimpan secara internal dengan Amazon Fraud Detector disimpan di tingkat`Event Type` sumber daya. Jadi, semua data acara yang berasal dari acara yang sama disimpan dalam satu`Event Type`. Acara yang disimpan nantinya dapat digunakan untuk melatih model baru atau melatih kembali model yang ada. Saat melatih model menggunakan data peristiwa yang disimpan, Anda dapat menentukan rentang waktu peristiwa secara opsional untuk membatasi ukuran set data latihan Anda.

Setiap kali Anda menyimpan data di Amazon Fraud Detector, menggunakan konsol Amazon Fraud Detector,`SendEvent` API, atau`CreateBatchImportJob` API, Amazon Fraud Detector memvalidasi data Anda sebelum menyimpan. Jika data Anda gagal validasi, data peristiwa tidak disimpan.

Prasyarat untuk menyimpan data secara internal dengan Amazon Fraud Detector

- Untuk memastikan bahwa data peristiwa Anda melewati validasi dan dataset berhasil disimpan, pastikan Anda telah menggunakan wawasan yang disediakan oleh Data models explorer untuk mempersiapkan dataset Anda.
- Membuat jenis peristiwa untuk data peristiwa yang ingin Anda simpan dengan Amazon Fraud Detector. Jika belum, ikuti intstructions untuk Membuat jenis peristiwa.

Validasi Data Cerdas

Saat Anda mengunggah kumpulan data di konsol Amazon Fraud Detector untuk impor batch, Amazon Fraud Detector menggunakan Smart Data Validation (SDV) untuk memvalidasi kumpulan data Anda sebelum mengimpor data Anda. SDV memindai file data yang diunggah dan mengidentifikasi masalah seperti data yang hilang, dan format atau tipe data yang salah. Selain memvalidasi kumpulan data Anda, SDV juga menyediakan laporan validasi yang mencantumkan semua masalah yang diidentifikasi dan menyarankan tindakan untuk memperbaiki masalah yang paling berdampak. Beberapa masalah yang diidentifikasi oleh SDV mungkin sangat penting dan harus diatasi sebelum Amazon Fraud Detector berhasil mengimpor kumpulan data Anda. Untuk informasi selengkapnya, lihat Laporan Validasi Data Cerdas.

SDV memvalidasi dataset Anda di tingkat file dan pada tingkat data (baris). Pada tingkat file, SDV memindai file data Anda dan mengidentifikasi masalah seperti izin yang tidak memadai untuk mengakses file, ukuran file yang salah, format file, dan header (metadata peristiwa dan variabel peristiwa). Pada tingkat data, SDV memindai setiap data peristiwa (baris) dan mengidentifikasi masalah seperti format data yang salah, panjang data, format stempel waktu, dan nilai null.

Validasi Data Cerdas saat ini hanya tersedia di konsol Amazon Fraud Detector dan validasi diaktifkan secara default. Jika Anda tidak ingin Amazon Fraud Detector menggunakan Validasi Data Cerdas sebelum mengimpor set data, matikan validasi di konsol Amazon Fraud Detector saat mengunggah set data Anda.

Memvalidasi data yang tersimpan saat menggunakan API atauAWS SDK

Saat mengunggah peristiwa melalui operasi`SendEvent`,`GetEventPrediction`, atau`CreateBatchImportJob` API, Amazon Fraud Detector memvalidasi hal-hal berikut:

- EventIngestion Pengaturan untuk jenis peristiwa tersebut DIAKTIFKAN.
- Timestamp peristiwa tidak dapat diperbarui. Peristiwa dengan ID peristiwa berulang dan EVENT_TIMESTAMP yang berbeda akan diperlakukan sebagai kesalahan.
- Nama dan nilai variabel sesuai dengan format yang diharapkan. Untuk informasi selengkapnya, lihat Buat variabel
- Variabel yang diperlukan diisi dengan nilai.
- Semua cap waktu acara tidak lebih dari 18 bulan dan tidak di future.

Menyimpan data acara menggunakan impor batch

Dengan fitur impor batch, Anda dapat dengan cepat dan mudah mengunggah kumpulan data peristiwa historis besar di Amazon Fraud Detector menggunakan konsol, API, atau AWS SDK. Untuk menggunakan impor batch, buat file input dalam format CSV yang berisi semua data peristiwa Anda, unggah file CSV ke bucket Amazon S3, dan mulai pekerjaan Impor. Amazon Fraud Detector pertama-tama memvalidasi data berdasarkan jenis peristiwa, dan kemudian secara otomatis mengimpor seluruh kumpulan data. Setelah data diimpor, data siap digunakan untuk melatih model baru atau untuk melatih kembali model yang ada.

File input dan output

File CSV masukan harus berisi header yang cocok dengan variabel yang didefinisikan dalam jenis peristiwa terkait ditambah empat variabel wajib. Lihat Siapkan data acara untuk penyimpanan untuk

informasi selengkapnya. Ukuran maksimum file data input adalah 20 Gigabyte (GB), atau sekitar 50 juta peristiwa. Jumlah acara akan bervariasi menurut ukuran acara Anda. Jika pekerjaan impor berhasil, file output kosong. Jika impor tidak berhasil, file keluaran berisi log kesalahan.

Buat file CSV

Amazon Fraud Detector mengompor data hanya dari file yang berada dalam format nilai yang dipisahkan dengan CSV). Baris pertama file CSV Anda harus berisi header kolom yang sama persis dengan variabel yang ditentukan dalam jenis peristiwa terkait ditambah empat variabel wajib: EVENT_ID, EVENT_TIMESTAMP, ENTITY_ID, dan ENTITY_TYPE. Anda juga dapat secara opsional menyertakan EVENT_LABEL dan LABEL_TIMESTAMP (LABEL_TIMESTAMP diperlukan jika EVENT_LABEL disertakan).

Tentukan variabel wajib

variabel wajib dianggap sebagai metadata peristiwa dan mereka harus ditentukan dalam huruf besar. Metadata peristiwa secara otomatis disertakan untuk pelatihan model. Tabel berikut berisi daftar variabel wajib, deskripsi masing-masing variabel, dan format yang diperlukan untuk variabel.

Nama	Penjelasan	Persyaratan
EVENT_ID	Pengidentifikasi untuk peristiwa tersebut. Misalnya, jika acara Anda adalah transaksi online, EVENT_ID mungkin merupakan nomor referensi transaksi yang diberikan kepada pelanggan Anda.	• EVENT_ID diperlukan untuk pekerjaan impor batch. • Nama ini harus unik untuk peristiwa itu. • Ini harus mewakili informasi yang berarti bagi bisnis Anda. • Ini harus memenuhi pola ekspresi reguler (misalnya ,`^[0-9a-z_-]+$.`) • Kami tidak menyarankan Anda menambahkan stempel waktu ke EVENT_ID. Melakukan hal itu dapat menyebabkan masalah saat Anda

Nama	Penjelasan	Persyaratan
		memperbarui acara. Ini karena Anda harus memberikan EVENT_ID yang sama persis jika Anda melakukan ini.

Nama	Penjelasan	Persyaratan

EVENT_TIMESTAMP	Timestamp saat acara terjadi. Timestamp harus dalam standar ISO 8601 di UTC.	• EVENT_TIMESTAMP diperlukan untuk pekerjaan impor batch.
		• Ini harus ditentukan dalam salah satu format berikut:
		  • %YYY-%mm-%ddT%hh: %mm: %ssZ (standar ISO 8601 dalam UTC hanya tanpa milidetik)
		
		  Contoh: 2019-11-30T 13:01:01 Z
		  • %yyyy/%mm/%dd %hh: %mm: %ss (AM/PM)
		
		  Contoh: 2019/11/3 0 13:01:01 PM, atau 2019/11/30 13:01:01
		  • %mm/%dd/%yyyy %hh: %mm: %ss
		
		  Contoh: 30/11/2019 13:01:01 WIB, 30/11/2019 13:01:01
		  • %mm/%dd/%yy %hh: %mm: %ss
		
		  Contoh: 11/30/19 13:01:01 PM, 11/30/19 13:01:01
		• Amazon Fraud Detector membuat asumsi berikut saat mengurai format tanggal/stempel waktu untuk cap waktu peristiwa:

Nama	Penjelasan	Persyaratan

		• Jika Anda menggunakan standar ISO 8601, itu harus sama persis dengan spesifikasi sebelumnya
		• Jika Anda menggunakan salah satu format lain, ada fleksibilitas tambahan:
		  • Selama berbulan-bulan dan berhari-hari, Anda dapat memberikan satu atau dua digit. Misalnya, 1/12/2019 adalah tanggal yang valid.
		  • Anda tidak perlu menyertakan hh:mm:ss jika Anda tidak memilikinya (yaitu, Anda cukup memberikan tanggal). Anda juga dapat memberikan subset hanya jam dan menit (misalnya, hh:mm). Hanya menyediakan jam tidak mendukung. Milidetik juga tidak didukung.
		  • Jika Anda memberikan label AM/PM, jam 12 jam diasumsikan. Jika tidak ada informasi

Nama	Penjelasan	Persyaratan
		AM/PM, jam 24 jam diasumsikan. • Anda dapat menggunakan "/" atau "-" sebagai pembatas untuk elemen tanggal. ":" diasumsikan untuk elemen timestamp.
ENTITY_ID	Pengidentifikasi untuk entitas melakukan peristiwa tersebut.	• ENTITY_ID diperlukan untuk pekerjaan impor batch • Itu harus mengikuti pola ekspresi reguler:`^[0-9A-Za-z_.@+-]+$` . • Jika id entitas tidak tersedia pada saat evaluasi, tentukan id entitas sebagai tidak diketahui.
ENTITY_TYPE	Entitas yang melakukan acara, seperti pedagang atau pelanggan	ENTITY_TYPE diperlukan untuk pekerjaan impor batch
EVENT_LABEL	Mengklasifikasikan acara sebagai`fraudulent` atau`legitimate`	EVENT_LABEL diperlukan jika LABEL_TIMESTAMP disertakan
LABEL_TIMESTAMP	Stempel waktu saat label acara terakhir diisi atau diperbarui	• LABEL_TIMESTAMP diperlukan jika EVENT_LABEL disertakan. • Ini harus mengikuti format stempel waktu.

Unggah file CSV ke Amazon S3 untuk impor batch

Setelah Anda membuat file CSV dengan data Anda, unggah file ke bucket Amazon Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Storage Service (Amazon S3) Simple Simple

Untuk mengunggah data peristiwa ke bucket Amazon S3

1. Masuk ke AWS Management Console dan buka konsol Amazon S3 di https://console.aws.amazon.com/s3/.
2. Pilih Buat bucket.

   Buat bucket membuka wizard.
3. Di Nama bucket, masukkan nama yang sesuai dengan DNS untuk bucket Anda.

   Nama kelompok harus:

   - Unik di seluruh Amazon S3.
   - Panjangnya antara 3 dan 63 karakter.
   - Tidak mengandung karakter huruf besar.
   - Mulai dengan huruf atau angka kecil.

   Setelah membuat bucket, Anda tidak dapat mengubah namanya. Untuk informasi tentang penamaan, lihat Aturan penamaan bucket dalam Panduan Pengguna Amazon Simple Simple Simple Simple Simple Simple Simple Simple Simple S

   > **Important**
   >
   > Hindari memasukkan informasi sensitif, seperti nomor akun, dalam nama kelompok. Nama bucket terlihat dalam URL yang menunjuk objek dalam bucket.
4. Di Wilayah, pilihAWS Wilayah tempat Anda ingin bucket berada. Anda harus memilih Wilayah yang sama di mana Anda menggunakan Amazon Fraud Detector, US East (N. Virginia), US East (Ohio), US West (Oregon), Europe (Irlandia), Asia Pacific (Singapore) atau Asia Pacific (Sydney).
5. Pada Pengaturan bucket untuk Blokir Akses Publik, pilih pengaturan Blokir Akses Publik yang ingin Anda terapkan ke bucket.

   Kami menyarankan agar Anda membiarkan semua pengaturan diaktifkan. Untuk informasi selengkapnya tentang memblokir akses publik, lihat Memblokir akses publik ke penyimpanan

Amazon S3 Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple Simple S

6. Pilih Create bucket (Buat bucket).
7. Unggah file data pelatihan ke bucket Amazon S3 Anda. Perhatikan jalur lokasi Amazon S3 untuk file latihan Anda (misalnya, s3://bucketname/object.csv).

Data peristiwa impor Batch di konsol Amazon Fraud Detector

Anda dapat dengan mudah mengimpor sejumlah besar kumpulan data peristiwa Anda di konsol Amazon Fraud Detector, menggunakan`CreateBatchImportJob` API atau menggunakan AWS SDK. Sebelum melanjutkan, pastikan Anda telah mengikuti petunjuk untuk menyiapkan kumpulan data Anda sebagai file CSV. Pastikan Anda juga mengunggah file CSV ke bucket Amazon S3.

Menggunakan konsol Amazon Fraud Detector

Untuk batch mengimpor data peristiwa di konsol

1. Buka Konsol AWS dan masuk ke akun Anda, dan navigasikan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Peristiwa.
3. Pilih jenis peristiwa Anda.
4. Pilih tab Peristiwa yang disimpan.
5. Di panel Rincian peristiwa Tersimpan, pastikan bahwa konsumsi peristiwa AKTIF.
6. Dalam panel Impor data peristiwa, pilih Impor Baru.
7. Di halaman impor peristiwa baru, berikan informasi berikut:
   - [Disarankan] Tinggalkan Aktifkan Validasi Data Cerdas untuk kumpulan data ini - set baru ke pengaturan default.
   - Untuk peran IAM untuk data, pilih peran IAM yang Anda buat untuk bucket Amazon S3 yang menyimpan file CSV yang akan Anda impor.
   - Untuk lokasi data input, masukkan lokasi S3 di mana Anda memiliki file CSV Anda.
   - Jika Anda ingin menentukan lokasi terpisah untuk menyimpan hasil impor Anda, klik Pisahkan lokasi data untuk input dan hasil tombol dan berikan lokasi bucket Amazon S3 yang valid.

> **Important**
>
> Pastikan peran IAM yang Anda pilih memiliki izin baca ke bucket Amazon S3 input Anda dan izin tulis ke bucket Amazon S3 keluaran Anda.

8. Pilih Mulai.
9. Kolom Status di panel Data peristiwa Impor menampilkan status validasi dan pekerjaan impor Anda. Banner di bagian atas memberikan deskripsi tingkat tinggi status sebagai dataset Anda pertama melewati validasi dan kemudian impor.
10. Ikuti panduan yang diberikan kepadaMemantau kemajuan validasi set data dan pekerjaan impor.

Memantau kemajuan validasi set data dan pekerjaan impor

Jika Anda menggunakan konsol Amazon Fraud Detector untuk melakukan pekerjaan impor batch, secara default, Amazon Fraud Detector memvalidasi set data Anda sebelum diimpor. Anda dapat memantau kemajuan dan status validasi dan pekerjaan impor di halaman Impor peristiwa baru konsol Amazon Fraud Detector. Spanduk di bagian atas halaman memberikan deskripsi singkat tentang temuan validasi dan status pekerjaan impor. Bergantung pada temuan validasi dan status pekerjaan impor Anda, Anda mungkin diminta untuk mengambil tindakan untuk memastikan validasi dan impor kumpulan data yang berhasil.

Tabel berikut memberikan rincian tindakan yang harus Anda ambil tergantung pada hasil validasi dan operasi impor.

Pesan spanduk	Status	Apa artinya	Apa yang harus saya lakukan
Validasi data telah dimulai	Validasi sedang berlangsung	SDV telah mulai memvalidasi dataset Anda	Tunggu status berubah
Validasi data tidak dapat dilanjutkan karena kesalahan dalam kumpulan	Validasi gagal	SDV mengidentifikasi masalah	Di panel Impor data peristiwa, pilih Id Job dan lihat laporan validasi. Ikuti Rekomendasi dalam laporan untuk

Pesan spanduk	Status	Apa artinya	Apa yang harus saya lakukan
data Anda. Perbaiki kesalahan dalam file data Anda dan mulai pekerjaan impor baru. Lihat laporan validasi untuk informasi lebih lanjut		dalam file data Anda. Masalah ini harus diatasi agar berhasil mengimpor kumpulan data Anda.	mengatasi semua kesalahan yang tercantum. Untuk informasi selengkapnya, lihat Menggunakan laporan validasi.
Pengimpor data telah dimulai. Validasi berhasil diselesaikan	Mengimpor sedang berlangsung	Dataset Anda melewati validasi. AFD sudah mulai mengimpor dataset Anda	Tunggu status berubah
Validasi dilengkapi dengan peringatan. Impor data telah dimulai	Mengimpor sedang berlangsung	Beberapa data dalam dataset Anda gagal validasi. Namun, data yang lulus validasi memenuhi persyaratan ukuran data minimum untuk impor.	Pantau pesan di spanduk dan tunggu status berubah

Pesan spanduk	Status	Apa artinya	Apa yang harus saya lakukan
Data Anda sebagian diimpor. Beberapa data gagal validasi dan tidak mendapatk an impor. Lihat laporan validasi untuk informasi lebih lanjut.	Diimpor. Status menampilkan ikon peringatan.	Beberapa data dalam file data Anda yang gagal validasi tidak mendapatk an impor. Sisa data yang lulus validasi diimpor.	Di panel Impor data peristiwa , pilih Id Job dan lihat laporan validasi. Ikuti Rekomendasi di tabel peringatan tingkat data untuk mengatasi peringatan yang tercantum. Anda tidak perlu mengatasi semua peringatan. Namun, pastikan bahwa dataset Anda memiliki lebih dari 50% data yang lolos validasi untuk impor yang berhasil. Setelah Anda membahas peringatan, mulailah pekerjaan impor baru. Untuk informasi selengkapnya, lihat Menggunakan laporan validasi.
Impor data gagal karena kesalahan pemrosesan. Mulai pekerjaan impor data baru	Pengambilan gagal	Impor gagal karena kesalahan run-time sementara	Memulai pekerjaan impor baru
Data berhasil diimpor	Impor	Validasi dan impor berhasil diselesaikan	Pilih Id Job pekerjaan impor Anda untuk melihat detail dan kemudian lanjutkan dengan pelatihan model

> **Note**
>
> Sebaiknya tunggu 10 menit setelah dataset berhasil diimpor ke Amazon Fraud Detector untuk memastikan bahwa data tersebut tertelan sepenuhnya oleh sistem.

Laporan Validasi Data Cerdas

Validasi Data Cerdas membuat laporan validasi setelah validasi selesai. Laporan validasi memberikan rincian semua masalah yang telah diidentifikasi SDV dalam kumpulan data Anda, dengan tindakan yang disarankan untuk memperbaiki masalah yang paling berdampak. Anda dapat menggunakan laporan validasi untuk menentukan apa masalahnya, di mana masalah berada di kumpulan data, tingkat keparahan masalah, dan cara memperbaikinya. Laporan validasi dibuat bahkan ketika validasi selesai dengan sukses. Dalam kasus ini, Anda dapat melihat laporan untuk melihat apakah ada masalah yang tercantum dan jika ada, putuskan apakah Anda ingin memperbaikinya.

> **Note**
>
> Versi SDV saat ini memindai kumpulan data Anda untuk masalah yang mungkin menyebabkan impor batch gagal. Jika validasi dan impor batch berhasil, kumpulan data Anda masih dapat mengalami masalah yang mungkin menyebabkan pelatihan model gagal. Kami menyarankan Anda untuk melihat laporan validasi meskipun validasi dan impor berhasil, dan mengatasi masalah apa pun yang tercantum dalam laporan untuk pelatihan model yang berhasil. Setelah Anda mengatasi masalah, buat pekerjaan impor batch baru.

Mengakses laporan validasi

Anda dapat mengakses laporan validasi kapan saja setelah validasi selesai menggunakan salah satu opsi berikut:

1. Setelah validasi selesai dan saat pekerjaan impor sedang berlangsung, di spanduk atas, pilih Lihat laporan validasi.
2. Setelah Job impor selesai, di panel Impor data peristiwa, pilih ID Pekerjaan dari pekerjaan impor yang baru saja selesai.

Menggunakan laporan validasi

Halaman laporan validasi pekerjaan impor Anda memberikan rincian pekerjaan impor ini, daftar kesalahan kritis jika ditemukan, daftar peringatan tentang peristiwa tertentu (baris) dalam kumpulan data Anda jika ditemukan, dan ringkasan singkat kumpulan data Anda yang mencakup informasi seperti nilai yang tidak valid, dan nilai yang hilang untuk setiap variabel.

- Impor detail pekerjaan

Memberikan rincian pekerjaan impor. Jika pekerjaan impor Anda gagal atau kumpulan data Anda diimpor sebagian, pilih Buka file hasil untuk melihat log kesalahan peristiwa yang gagal diimpor.

- Kesalahan kritis

  Memberikan rincian masalah yang paling berdampak dalam kumpulan data Anda yang diidentifikasi oleh SDV. Semua masalah yang tercantum dalam panel ini sangat penting dan Anda harus mengatasinya sebelum melanjutkan impor. Jika Anda mencoba mengimpor kumpulan data tanpa mengatasi masalah kritis, pekerjaan impor Anda mungkin gagal.

  Untuk mengatasi masalah kritis, ikuti rekomendasi yang diberikan untuk setiap peringatan. Setelah Anda mengatasi semua masalah yang tercantum di panel Kesalahan kritis, buat pekerjaan impor batch baru.

- Peringatan tingkat data

  Menyediakan ringkasan peringatan untuk peristiwa tertentu (baris) dalam kumpulan data Anda. Jika panel peringatan tingkat data diisi, beberapa peristiwa di set data Anda gagal validasi dan tidak diimpor.

  Untuk setiap peringatan, kolom Deskripsi menampilkan jumlah peristiwa yang memiliki masalah. Dan ID kejadian Contoh menyediakan sebagian daftar contoh ID kejadian yang dapat Anda gunakan sebagai titik awal untuk menemukan sisa peristiwa yang memiliki masalah. Gunakan Rekomendasi yang disediakan untuk peringatan untuk memperbaiki masalah. Juga gunakan log kesalahan dari file keluaran Anda untuk informasi tambahan tentang masalah ini. Log kesalahan yang dihasilkan untuk semua peristiwa yang gagal batch impor. Untuk mengakses log kesalahan, di panel Import job details, pilih Buka file hasil.

  > **Note**
  >
  > Jika lebih dari 50% peristiwa (baris) dalam dataset Anda gagal validasi, pekerjaan impor juga gagal. Dalam hal ini, Anda harus memperbaiki data sebelum memulai pekerjaan impor baru.

- Ringkasan set data

  Menyediakan ringkasan laporan validasi kumpulan data Anda. Jika kolom Jumlah peringatan menunjukkan lebih dari peringatan 0, putuskan apakah Anda perlu memperbaiki peringatan tersebut. Jika kolom Jumlah peringatan menunjukkan 0 detik, lanjutkan untuk melatih model Anda.

Data peristiwa impor Batch menggunakan AWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan sampel untuk CreateBatchImportJobAPI. Pekerjaan impor batch harus menyertakan JobID, InputPath, OutputPath, eventTypeNamedan iamRoleArn. JobID tidak dapat berisi ID yang sama dari pekerjaan sebelumnya, kecuali pekerjaan itu ada dalam status CREATE_FAILED. InputPath dan OutputPath harus berupa jalur S3 yang valid. Anda dapat memilih untuk tidak menentukan nama file di OutputPath, namun, Anda masih perlu menyediakan lokasi bucket S3 yang valid. eventTypeName Dan iamRoleArn harus ada. Peran IAM harus memberikan izin baca untuk memasukkan bucket Amazon S3 dan izin menulis untuk mengeluarkan bucket Amazon S3.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_batch_import_job (
jobId = 'sample_batch_import',
inputPath = 's3://bucket_name/input_file_name.csv',
outputPath = 's3://bucket_name/',
eventTypeName = 'sample_registration',
iamRoleArn: 'arn:aws:iam::************:role/service-role/AmazonFraudDetector-
DataAccessRole-*************'
)
```

Batalkan pekerjaan impor batch

Anda dapat membatalkan pekerjaan impor batch yang sedang berlangsung kapan saja di konsol Amazon Fraud Detector, menggunakan`CancelBatchImportJob` API, atau AWS SDK.

Untuk membatalkan pekerjaan impor batch di konsol,

1. Buka Konsol AWS dan masuk ke akun Anda, dan navigasikan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Peristiwa.
3. Pilih jenis peristiwa Anda.
4. Pilih tab Peristiwa yang disimpan.
5. Di panel Impor data peristiwa, pilih Id pekerjaan dari pekerjaan impor dalam proses yang ingin Anda batalkan.
6. Di halaman pekerjaan acara, klik Tindakan dan pilih Batalkan impor peristiwa.
7. Pilih Hentikan impor peristiwa untuk membatalkan pekerjaan impor batch.

Membatalkan pekerjaan impor batch menggunakan AWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan sampel untuk`CancelBatchImportJob` API. Pekerjaan pembatalan impor harus menyertakan ID pekerjaan dari pekerjaan impor batch yang sedang berlangsung.

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraudDetector.cancel_batch_import_job (
    jobId = 'sample_batch'
)
```

Menyimpan data peristiwa menggunakan operasi GetEventPredictions API

Secara default, semua peristiwa yang dikirim ke`GetEventPrediction` API untuk evaluasi disimpan di Amazon Fraud Detector. Ini berarti bahwa Amazon Fraud Detector akan secara otomatis menyimpan data peristiwa ketika Anda membuat prediksi dan menggunakan data tersebut untuk memperbarui variabel yang dihitung dalam waktu hampir nyata. Anda dapat menonaktifkan penyimpanan data dengan menavigasi ke jenis peristiwa di konsol Amazon Fraud Detector dan mengatur konsumsi peristiwa OFF atau memperbarui EventIngestion nilai ke DISABLED menggunakan operasi`PutEventType` API. Untuk informasi selengkapnya tentang operasi`GetEventPrediction` API, lihatPrediksi penipuan.

> **Important**
>
> Kami sangat menyarankan bahwa setelah Anda mengaktifkan Event ingestion untuk jenis Event, tetap aktifkan. Menonaktifkan konsumsi peristiwa untuk jenis Peristiwa yang sama dan kemudian menghasilkan prediksi dapat mengakibatkan perilaku yang tidak konsisten.

Menyimpan data peristiwa menggunakan operasi SendEvent API

Anda dapat menggunakan operasi`SendEvent` API untuk menyimpan peristiwa di Amazon Fraud Detector tanpa menghasilkan prediksi penipuan untuk kejadian tersebut. Misalnya, Anda dapat menggunakan`SendEvent` operasi untuk mengunggah kumpulan data historis, yang nantinya dapat Anda gunakan untuk melatih model.

Format Timestamp Peristiwa untuk SendEvent API

Saat menyimpan data peristiwa menggunakan`SendEvent` API, Anda harus memastikan bahwa stempel waktu acara Anda dalam format yang diperlukan. Amazon Fraud Detector mendukung format tanggal/cap waktu berikut:

- %YYY-%mm-%ddT%hh: %mm: %ssZ (standar ISO 8601 dalam UTC hanya tanpa milidetik)

  Contoh: 2019-11-30T 13:01:01 Z
- %yyyy/%mm/%dd %hh: %mm: %ss (AM/PM)

  Contoh: 2019/11/30 13:01:01 PM, atau 2019/11/30 13:01:01
- %mm/%dd/%yyyy %hh: %mm: %ss

  Contoh: 30/11/2019 13:01:01 WIB, 30/11/2019 13:01:01
- %mm/%dd/%yy %hh: %mm: %ss

  Contoh: 11/30/19 13:01:01 PM, 11/30/19 13:01:01

Amazon Fraud Detector membuat asumsi berikut saat mengurai format tanggal/stempel waktu untuk cap waktu peristiwa:

- Jika Anda menggunakan standar ISO 8601, itu harus sama persis dengan spesifikasi sebelumnya
- Jika Anda menggunakan salah satu format lain, ada fleksibilitas tambahan:
  - Selama berbulan-bulan dan berhari-hari, Anda dapat memberikan satu atau dua digit. Misalnya, 1/12/2019 adalah tanggal yang valid.
  - Anda tidak perlu menyertakan hh:mm:ss jika Anda tidak memilikinya (yaitu, Anda cukup memberikan tanggal). Anda juga dapat memberikan subset hanya jam dan menit (misalnya, hh:mm). Hanya menyediakan jam tidak mendukung. Milidetik juga tidak didukung.
  - Jika Anda memberikan label AM/PM, jam 12 jam diasumsikan. Jika tidak ada informasi AM/PM, jam 24 jam diasumsikan.
  - Anda dapat menggunakan “/” atau “-” sebagai pembatas untuk elemen tanggal. “:” diasumsikan untuk elemen timestamp.

Berikut ini adalah contoh panggilan`SendEvent` API.

```
import boto3
```

```
fraudDetector = boto3.client('frauddetector')

fraudDetector.send_event(
            eventId        = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
            eventTypeName  = 'sample_registration',
            eventTimestamp = '2020-07-13T23:18:21Z',
            eventVariables =  {
       'email_address' : 'johndoe@exampledomain.com',
       'ip_address' : '1.2.3.4'},
            assignedLabel  = 'legit',
            labelTimestamp = '2020-07-13T23:18:21Z',
            entities       = [{'entityType':'sample_customer', 'entityId':'12345'}],

)
```

Dapatkan detail data peristiwa yang disimpan

Setelah menyimpan data peristiwa di Amazon Fraud Detector, Anda dapat memeriksa data terbaru yang disimpan untuk suatu peristiwa menggunakan GetEventAPI. Contoh kode berikut memeriksa data terbaru yang disimpan untuk`sample_registration` acara tersebut.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.get_event(
            eventId        = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
            eventTypeName  = 'sample_registration'
)
```

Melihat metrik kumpulan data peristiwa yang disimpan

Untuk setiap jenis peristiwa, Anda dapat melihat metrik seperti, jumlah peristiwa yang disimpan, ukuran total peristiwa yang disimpan, dan stempel waktu peristiwa tersimpan paling awal dan terbaru, di konsol Amazon Fraud Detector.

Untuk melihat metrik peristiwa yang disimpan dari jenis peristiwa,

1. BukaAWS Konsol dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.

2. Di panel navigasi kiri, pilih Peristiwa.
3. Pilih jenis peristiwa Anda.
4. Pilih tab Acara tersimpan.
5. Panel Detail peristiwa Tersimpan menampilkan metrik. Metrik ini diperbarui secara otomatis satu kali per hari.
6. Secara opsional klik Segarkan metrik peristiwa untuk memperbarui metrik Anda secara manual.

> **Note**
>
> Jika Anda baru saja mengimpor data, sebaiknya tunggu 5 - 10 menit setelah Anda selesai mengimpor data untuk menyegarkan dan melihat metrik.

Orkestrasi acara

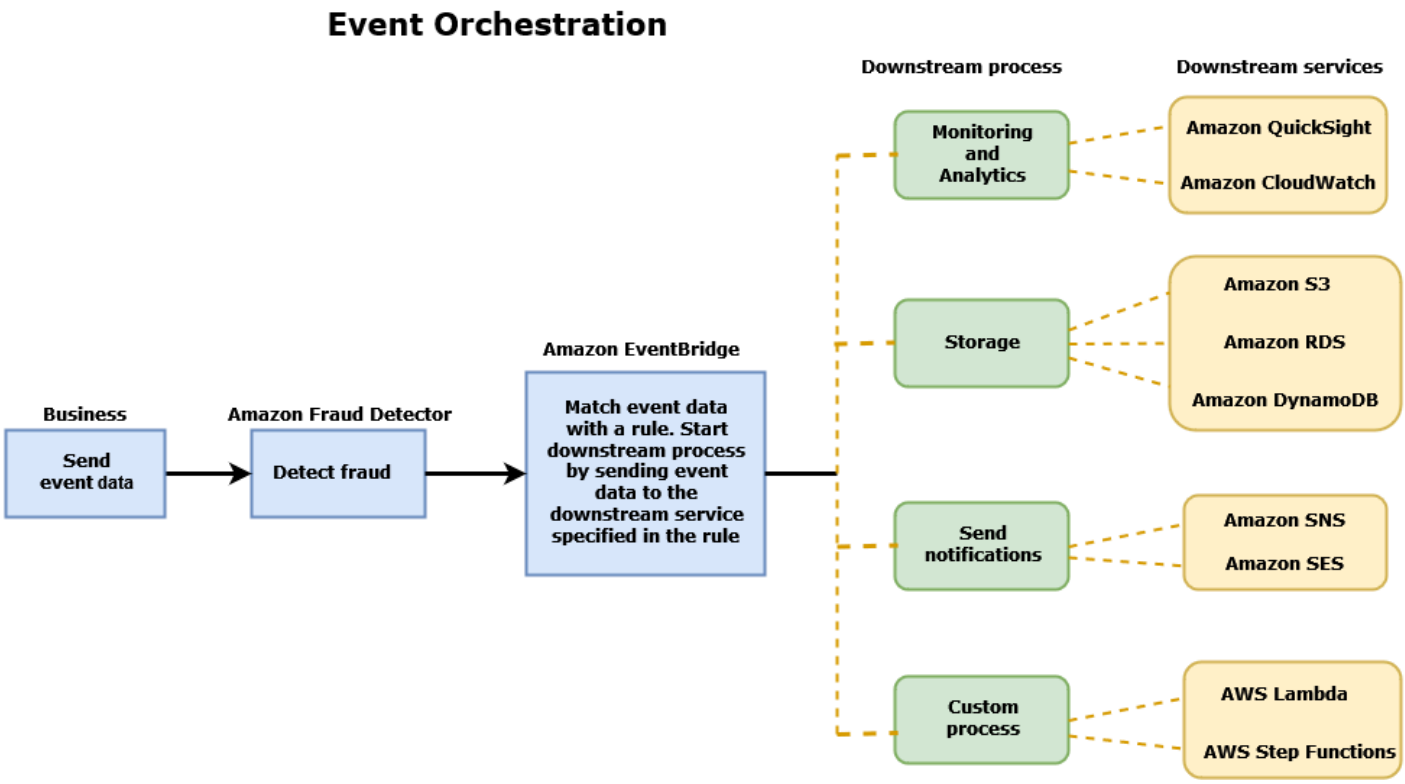
Orkestrasi acara memudahkan Anda mengirim acara Layanan AWS untuk pemrosesan hilir, menggunakan Amazon. EventBridge Amazon Fraud Detector memberi Anda aturan sederhana yang dapat Anda gunakan untuk mengotomatiskan pemrosesan peristiwa setelah deteksi penipuan. Dengan orkestrasi peristiwa, Anda dapat mengotomatiskan proses peristiwa hilir seperti, mengirim acara ke dasbor untuk mendapatkan wawasan dari data peristiwa, menghasilkan pemberitahuan berdasarkan hasil deteksi penipuan, dan memperbarui peristiwa dengan label berdasarkan pembelajaran dari deteksi penipuan.

Orkestrasi acara menyediakan akses mudah ke layanan di AWS lingkungan, melalui Amazon. EventBridge Anda dapat mengonfigurasi Amazon EventBridge untuk mengirim peristiwa secara langsung ke Layanan AWS atau tidak langsung menggunakan tujuan API. Yang Layanan AWS Anda gunakan untuk mengatur proses hilir Anda juga disebut target. Beberapa target yang dapat Anda gunakan untuk mengatur pemrosesan hilir adalah sebagai berikut:

- Untuk pemantauan dan analitik - Amazon QuickSight, Amazon CloudWatch
- Untuk penyimpanan - Amazon S3, AmazonRDS, Amazon DynamoDB
- Untuk mengirim pemberitahuan - Amazon SNS, Amazon SES
- Untuk pemrosesan khusus — AWS Lambda, AWS Step Functions

Untuk informasi selengkapnya tentang target orkestrasi yang didukung oleh Amazon, EventBridge lihat Target Amazon. EventBridge

Diagram berikut memberikan pandangan tingkat tinggi tentang cara kerja orkestrasi acara.



Menyiapkan orkestrasi acara

Menyiapkan orkestrasi peristiwa untuk acara Anda mengharuskan Anda menyiapkan proses di layanan target, mengonfigurasi Amazon EventBridge untuk menerima dan mengirim data peristiwa, dan membuat aturan di Amazon EventBridge yang menentukan kondisi untuk memulai proses hilir. Selesaikan langkah-langkah berikut untuk mengatur orkestrasi acara:

Untuk mengatur orkestrasi acara

1. Buka Panduan EventBridge Pengguna Amazon dan pelajari cara menggunakan Amazon EventBridge. Pastikan untuk mempelajari cara membuat Aturan di Amazon EventBridge untuk kasus penggunaan Anda.
2. Ikuti instruksi untukAktifkan orkestrasi peristiwa di Amazon Fraud Detector.

> **Note**
>
> Orkestrasi acara untuk acara Anda dinonaktifkan secara default.

3. Siapkan layanan target Anda untuk menerima dan memproses data acara. Misalnya, jika proses hilir Anda melibatkan pengiriman notifikasi dan Anda ingin menggunakan Amazon SNS, buka konsol Amazon SNS, buat topik SNS, lalu berlangganan titik akhir ke topik tersebut.
4. Ikuti petunjuk untuk membuat EventBridge aturan Amazon.

> **Important**
>
> Saat membuat pola acara di Amazon EventBridge, pastikan `aws.frauddetector` untuk menyediakan bidang sumber dan `Event Prediction Result Returned` bidang tipe detail.

Aktifkan orkestrasi peristiwa di Amazon Fraud Detector

Anda dapat mengaktifkan orkestrasi acara untuk suatu acara baik saat Anda membuat jenis acara atau setelah Anda membuat jenis acara Anda. Orkestrasi peristiwa dapat diaktifkan di konsol Amazon Fraud Detector, menggunakan `put-event-type` perintah, menggunakan `PutEventType` API, atau menggunakan. AWS SDK for Python (Boto3)

Aktifkan orkestrasi peristiwa di konsol Amazon Fraud Detector

Contoh ini memungkinkan orkestrasi acara untuk jenis acara yang telah dibuat. Jika Anda membuat jenis acara baru dan ingin mengaktifkan orkestrasi, ikuti instruksi untuk. Membuat jenis acara

Untuk mengaktifkan orkestrasi acara

1. Buka Konsol AWS Manajemen dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Acara.
3. Di halaman Jenis acara, pilih jenis acara Anda.
4. Aktifkan Aktifkan orkestrasi acara dengan Amazon. EventBridge
5. Lanjutkan dengan petunjuk langkah 3 untukMenyiapkan orkestrasi acara.

Aktifkan orkestrasi acara menggunakan AWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan sampel untuk memperbarui jenis acara `sample_registration` untuk mengaktifkan orkestrasi acara. Contoh menggunakan `PutEventType` API dan mengasumsikan Anda telah membuat variabel `ip_address`

danemail_address, label legit danfraud, dan jenis sample_customer entitas. Untuk informasi tentang cara membuat sumber daya ini, lihat Sumber daya.

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraud_detector.put_event_type(
  name = 'sample_registration',
  eventVariables = ['ip_address', 'email_address'],
  eventOrchestration = {'eventBridgeEnabled': True},
  labels = ['legit', 'fraud'],
  entityTypes = ['sample_customer'])
```

Nonaktifkan orkestrasi acara di Amazon Fraud Detector

Anda dapat menonaktifkan orkestrasi peristiwa untuk suatu peristiwa kapan saja di konsol Amazon Fraud Detector, menggunakan put-event-type perintah, menggunakan PutEventType API, atau menggunakan. AWS SDK for Python (Boto3)

Nonaktifkan orkestrasi peristiwa di konsol Amazon Fraud Detector

Untuk menonaktifkan orkestrasi acara

1. Buka Konsol AWS Manajemen dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Acara.
3. Di halaman Jenis acara, pilih jenis acara Anda.
4. Matikan Aktifkan orkestrasi acara dengan Amazon. EventBridge

Nonaktifkan orkestrasi acara menggunakan AWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan sampel untuk memperbarui jenis acara sample_registration untuk menonaktifkan orkestrasi acara menggunakan API. PutEventType

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraud_detector.put_event_type(
  name = 'sample_registration',
  eventVariables = ['ip_address', 'email_address'],
  eventOrchestration = {'eventBridgeEnabled': False},
```

```
  entityTypes = ['sample_customer'])
```

Model

Amazon Fraud Detector menggunakan model pembelajaran mesin untuk menghasilkan prediksi penipuan. Setiap model dilatih menggunakan tipe model. Jenis model menentukan algoritma dan transformasi yang digunakan untuk melatih model. Pelatihan model adalah proses menggunakan kumpulan data yang Anda berikan untuk membuat model yang dapat memprediksi peristiwa penipuan.

Untuk membuat model, Anda harus terlebih dahulu memilih jenis model dan kemudian menyiapkan dan memberikan data yang akan digunakan untuk melatih model.

Pilih jenis model

Jenis model berikut tersedia di Amazon Fraud Detector. Pilih jenis model yang sesuai untuk kasus penggunaan Anda.

- Wawasan Penipuan Online

  Jenis model Wawasan Penipuan Online dioptimalkan untuk mendeteksi penipuan ketika sedikit data historis tersedia tentang entitas yang sedang dievaluasi, misalnya, pelanggan baru yang mendaftar secara online untuk akun baru.
- Wawasan Penipuan Transaksi

  Jenis model Transaction Fraud Insights paling cocok untuk mendeteksi kasus penggunaan penipuan di mana entitas yang sedang dievaluasi mungkin memiliki riwayat interaksi yang dapat dianalisis model untuk meningkatkan akurasi prediksi (misalnya, pelanggan yang sudah ada dengan riwayat pembelian sebelumnya).
- Wawasan Pengambilalihan Akun

  Jenis model Account Takeover Insights mendeteksi jika akun dikompromikan oleh phishing atau jenis serangan lainnya. Data login dari akun yang disusupi, seperti browser dan perangkat yang digunakan saat login, berbeda dari data login historis yang terkait dengan akun.

Wawasan penipuan online

Online Fraud Insights adalah model pembelajaran mesin yang diawasi, yang berarti menggunakan contoh historis transaksi penipuan dan sah untuk melatih model tersebut. Model Online Fraud

Insights dapat mendeteksi penipuan berdasarkan sedikit data historis. Masukan model fleksibel, sehingga Anda dapat menyesuaikannya untuk mendeteksi berbagai risiko penipuan termasuk ulasan palsu, penyalahgunaan promosi, dan penipuan checkout tamu.

Model Online Fraud Insights menggunakan ansambel algoritma pembelajaran mesin untuk pengayaan data, transformasi, dan klasifikasi penipuan. Sebagai bagian dari proses pelatihan model, Online Fraud Insights memperkaya elemen data mentah seperti alamat IP dan nomor BIN dengan data pihak ketiga seperti geolokasi alamat IP atau bank penerbit untuk kartu kredit. Selain data pihak ketiga, Online Fraud Insights menggunakan algoritma pembelajaran mendalam yang memperhitungkan pola penipuan yang telah terlihat di Amazon dan. AWS Pola penipuan ini menjadi fitur masukan untuk model Anda menggunakan algoritme peningkatan pohon gradien.

Untuk meningkatkan kinerja, Online Fraud Insights mengoptimalkan parameter hiper dari algoritma peningkatan pohon gradien melalui proses optimasi Bayesian. Ini secara berurutan melatih lusinan model yang berbeda dengan parameter model yang bervariasi (seperti jumlah pohon, kedalaman pohon, dan jumlah sampel per daun). Ini juga menggunakan strategi optimasi yang berbeda seperti meningkatkan populasi penipuan minoritas untuk menjaga tingkat penipuan yang sangat rendah.

Memilih sumber data

Saat melatih model Wawasan Penipuan Online, Anda dapat memilih untuk melatih model pada data peristiwa yang disimpan secara eksternal (di luar Amazon Fraud Detector) atau disimpan dalam Amazon Fraud Detector. Penyimpanan eksternal yang saat ini didukung Amazon Fraud Detector adalah Amazon Simple Storage Service (Amazon S3). Jika Anda menggunakan penyimpanan eksternal, kumpulan data acara harus diunggah sebagai format nilai yang dipisahkan koma (CSV) ke bucket Amazon S3. Opsi penyimpanan data ini disebut dalam konfigurasi pelatihan model sebagai EXTERNAL_EVENTS (untuk penyimpanan eksternal) dan INGESTED_EVENTS (untuk penyimpanan internal). Untuk informasi selengkapnya tentang sumber data yang tersedia dan cara menyimpan data di dalamnya, lihatPenyimpanan data peristiwa.

Mempersiapkan data

Di mana pun Anda memilih untuk menyimpan data acara Anda (Amazon S3 atau Amazon Fraud Detector), persyaratan untuk jenis model Wawasan Penipuan Online adalah sama.

Dataset Anda harus berisi header kolom EVENT_LABEL. Variabel ini mengklasifikasikan suatu peristiwa sebagai penipuan atau sah. Saat menggunakan file CSV (penyimpanan eksternal), Anda harus menyertakan EVENT_LABEL untuk setiap peristiwa dalam file. Untuk penyimpanan internal, bidang EVENT_LABEL bersifat opsional tetapi semua peristiwa harus diberi label untuk disertakan

dalam kumpulan data pelatihan. Saat mengonfigurasi pelatihan model, Anda dapat memilih apakah akan mengabaikan peristiwa yang tidak berlabel, mengambil label yang sah untuk peristiwa yang tidak berlabel, atau mengasumsikan label penipuan untuk semua peristiwa yang tidak berlabel.

Memilih data

Lihat Mengumpulkan data acara untuk informasi tentang memilih data untuk melatih model Wawasan Penipuan Online Anda.

Pelatihan Online Fraud Insights memproses sampel dan mempartisi data historis berdasarkan EVENT_TIMESTAMP. Tidak perlu mengambil sampel data secara manual dan melakukannya dapat berdampak negatif pada hasil model Anda.

Variabel peristiwa

Model Wawasan Penipuan Online membutuhkan setidaknya dua variabel, terlepas dari metadata peristiwa yang diperlukan, yang telah lulus validasi data untuk pelatihan model dan memungkinkan hingga 100 variabel per model. Umumnya, semakin banyak variabel yang Anda berikan, semakin baik model dapat membedakan antara penipuan dan peristiwa yang sah. Meskipun model Wawasan Penipuan Online dapat mendukung lusinan variabel, termasuk variabel khusus, kami merekomendasikan menyertakan alamat IP dan alamat email karena variabel ini biasanya paling efektif dalam mengidentifikasi entitas yang sedang dievaluasi.

Memvalidasi data

Sebagai bagian dari proses pelatihan, Wawasan Penipuan Online akan memvalidasi kumpulan data untuk masalah kualitas data yang dapat memengaruhi pelatihan model. Setelah memvalidasi data, Amazon Fraud Detector akan mengambil tindakan yang tepat untuk membangun model terbaik. Ini termasuk mengeluarkan peringatan untuk masalah kualitas data potensial, secara otomatis menghapus variabel yang memiliki masalah kualitas data, atau mengeluarkan kesalahan dan menghentikan proses pelatihan model. Untuk informasi selengkapnya, lihat validasi kumpulan data.

Wawasan penipuan transaksi

Jenis model Transaction Fraud Insights dirancang untuk mendeteksi penipuan online, atau card-not-present, transaksi. Transaction Fraud Insights adalah model pembelajaran mesin yang diawasi, yang berarti menggunakan contoh historis transaksi penipuan dan sah untuk melatih model tersebut.

Model Transaction Fraud Insights menggunakan ansambel algoritma pembelajaran mesin untuk pengayaan data, transformasi, dan klasifikasi penipuan. Ini memanfaatkan mesin rekayasa fitur untuk

membuat agregat tingkat entitas dan tingkat peristiwa. Sebagai bagian dari proses pelatihan model, Transaction Fraud Insights memperkaya elemen data mentah seperti alamat IP dan nomor BIN dengan data pihak ketiga seperti geolokasi alamat IP atau bank penerbit untuk kartu kredit. Selain data pihak ketiga, Transaction Fraud Insights menggunakan algoritme pembelajaran mendalam yang memperhitungkan pola penipuan yang telah terlihat di Amazon dan Pola penipuan AWS ini menjadi fitur input untuk model Anda menggunakan algoritme peningkatan pohon gradien.

Untuk meningkatkan kinerja, Transaction Fraud Insights mengoptimalkan parameter hiper algoritme peningkatan pohon gradien melalui proses optimasi Bayesian, secara berurutan melatih lusinan model berbeda dengan parameter model yang bervariasi (seperti jumlah pohon, kedalaman pohon, jumlah sampel per daun) serta strategi pengoptimalan yang berbeda seperti meningkatkan populasi penipuan minoritas untuk menangani tingkat penipuan yang sangat rendah.

Sebagai bagian dari proses pelatihan model, mesin rekayasa fitur model Transaction Fraud menghitung nilai untuk setiap entitas unik dalam kumpulan data pelatihan Anda untuk membantu meningkatkan prediksi penipuan. Misalnya, selama proses pelatihan, Amazon Fraud Detector menghitung dan menyimpan terakhir kali entitas melakukan pembelian dan memperbarui nilai ini secara dinamis setiap kali Anda memanggil `GetEventPrediction` atau `SendEvent` API. Selama prediksi penipuan, variabel peristiwa digabungkan dengan entitas lain dan metadata peristiwa untuk memprediksi apakah transaksi tersebut curang.

Memilih sumber data

Model Transaction Fraud Insights dilatih pada kumpulan data yang disimpan secara internal hanya dengan Amazon Fraud Detector (INGESTED_EVENTS). Hal ini memungkinkan Amazon Fraud Detector untuk terus memperbarui nilai terhitung tentang entitas yang Anda evaluasi. Untuk informasi selengkapnya tentang sumber data yang tersedia, lihat Penyimpanan data peristiwa

Mempersiapkan data

Sebelum Anda melatih model Transaction Fraud Insights, pastikan bahwa file data Anda berisi semua header seperti yang disebutkan dalam Siapkan dataset acara. Model Transaction Fraud Insights membandingkan entitas baru yang diterima dengan contoh entitas penipuan dan sah dalam kumpulan data, sehingga sangat membantu untuk memberikan banyak contoh untuk setiap entitas.

Amazon Fraud Detector secara otomatis mengubah kumpulan data peristiwa yang disimpan menjadi format yang benar untuk pelatihan. Setelah model menyelesaikan pelatihan, Anda dapat meninjau metrik kinerja dan menentukan apakah Anda harus menambahkan entitas ke kumpulan data pelatihan Anda.

Memilih data

Secara default, Insights Penipuan Transaksi melatih seluruh kumpulan data yang disimpan untuk Jenis Peristiwa yang Anda pilih. Anda dapat secara opsional mengatur rentang waktu untuk mengurangi peristiwa yang digunakan untuk melatih model Anda. Saat menetapkan rentang waktu, pastikan bahwa catatan yang digunakan untuk melatih model memiliki waktu yang cukup untuk matang. Artinya, cukup waktu telah berlalu untuk memastikan catatan yang sah dan penipuan telah diidentifikasi dengan benar. Misalnya, untuk penipuan tolak bayar, seringkali dibutuhkan 60 hari atau lebih untuk mengidentifikasi peristiwa penipuan dengan benar. Untuk kinerja model terbaik, pastikan bahwa semua catatan dalam kumpulan data pelatihan Anda sudah matang.

Tidak perlu memilih rentang waktu yang mewakili tingkat penipuan yang ideal. Amazon Fraud Detector secara otomatis mengambil sampel data Anda untuk mencapai keseimbangan antara tingkat penipuan, rentang waktu, dan jumlah entitas.

Amazon Fraud Detector mengembalikan kesalahan validasi selama pelatihan model jika Anda memilih rentang waktu yang tidak cukup acara untuk berhasil melatih model. Untuk kumpulan data yang disimpan, bidang EVENT_LABEL bersifat opsional, tetapi peristiwa harus diberi label untuk disertakan dalam kumpulan data pelatihan Anda. Saat mengonfigurasi pelatihan model, Anda dapat memilih apakah akan mengabaikan peristiwa yang tidak berlabel, mengambil label yang sah untuk peristiwa yang tidak berlabel, atau mengasumsikan label penipuan untuk peristiwa yang tidak berlabel.

Variabel peristiwa

Jenis peristiwa yang digunakan untuk melatih model harus berisi setidaknya 2 variabel, selain dari metadata peristiwa yang diperlukan, yang telah melewati validasi data dan dapat berisi hingga 100 variabel. Umumnya, semakin banyak variabel yang Anda berikan, semakin baik model dapat membedakan antara penipuan dan peristiwa yang sah. Meskipun model Transaction Fraud Insight dapat mendukung lusinan variabel, termasuk variabel kustom, kami menyarankan Anda menyertakan alamat IP, alamat email, jenis instrumen pembayaran, harga pesanan, dan BIN kartu.

Memvalidasi data

Sebagai bagian dari proses pelatihan, Transaction Fraud Insights memvalidasi kumpulan data pelatihan untuk masalah kualitas data yang mungkin memengaruhi pelatihan model. Setelah memvalidasi data, Amazon Fraud Detector mengambil tindakan yang tepat untuk membangun model terbaik. Ini termasuk mengeluarkan peringatan untuk masalah kualitas data potensial, secara otomatis menghapus variabel yang memiliki masalah kualitas data, atau mengeluarkan kesalahan

dan menghentikan proses pelatihan model. Untuk informasi selengkapnya, lihat Validasi kumpulan data.

Amazon Fraud Detector akan mengeluarkan peringatan tetapi terus melatih model jika jumlah entitas unik kurang dari 1.500 karena ini dapat memengaruhi kualitas data pelatihan. Jika Anda menerima peringatan, tinjau metrik kinerja.

Wawasan pengambilalihan akun

Jenis model Account Takeover Insights (ATI) mengidentifikasi aktivitas online penipuan dengan mendeteksi apakah akun dikompromikan melalui pengambilalihan berbahaya, phishing, atau dari kredensi yang dicuri. Account Takeover Insights adalah model pembelajaran mesin yang menggunakan acara login dari bisnis online Anda untuk melatih model tersebut.

Anda dapat menyematkan model Wawasan Pengambilalihan Akun yang terlatih dalam alur login waktu nyata Anda untuk mendeteksi apakah akun dikompromikan. Model ini menilai berbagai jenis otentikasi dan login. Mereka termasuk login aplikasi web, otentikasi berbasis API, dan single-sign-on (SSO). Untuk menggunakan model Account Takeover Insights, panggil GetEventPredictionAPI setelah kredensyal login yang valid ditampilkan. API menghasilkan skor yang mengukur risiko akun dikompromikan. Amazon Fraud Detector menggunakan skor dan aturan yang Anda tetapkan untuk mengembalikan satu atau beberapa hasil untuk peristiwa login. Hasilnya adalah yang Anda konfigurasikan. Berdasarkan hasil yang Anda terima, Anda dapat mengambil tindakan yang tepat untuk setiap login. Artinya, Anda dapat menyetujui atau menantang kredensyal yang disajikan untuk login. Misalnya, Anda dapat menantang kredensialnya dengan meminta PIN akun sebagai verifikasi tambahan.

Anda juga dapat menggunakan model Account Takeover Insights untuk mengevaluasi login akun secara asinkron dan mengambil tindakan pada akun berisiko tinggi. Misalnya, akun berisiko tinggi dapat ditambahkan ke antrian investigasi untuk peninjau manusia untuk menentukan apakah tindakan lebih lanjut perlu diambil, seperti menangguhkan akun.

Model Account Takeover Insights dilatih menggunakan kumpulan data yang berisi peristiwa login historis bisnis Anda. Anda memberikan data ini. Anda dapat secara opsional melabeli akun sebagai sah atau curang. Namun, ini tidak diperlukan untuk melatih model. Model Account Takeover Insights mendeteksi anomali berdasarkan riwayat login akun yang berhasil. Ini juga mempelajari cara mendeteksi anomali dalam perilaku pengguna yang menunjukkan peningkatan risiko peristiwa pengambilalihan akun berbahaya. Misalnya, pengguna yang biasanya masuk dari perangkat dan alamat IP yang sama. Penipu biasanya masuk dari perangkat dan geolokasi yang berbeda. Teknik ini

menghasilkan skor risiko dari suatu aktivitas yang anomali, yang biasanya merupakan karakteristik utama dari pengambilalihan akun berbahaya.

Sebelum melatih model Account Takeover Insights, Amazon Fraud Detector menggunakan kombinasi teknik pembelajaran mesin untuk melakukan pengayaan data, agregasi data, dan transformasi data. Kemudian, selama proses pelatihan, Amazon Fraud Detector memperkaya elemen data mentah yang Anda berikan. Contoh elemen data mentah termasuk alamat IP dan agen pengguna. Amazon Fraud Detector menggunakan elemen-elemen ini untuk membuat input tambahan yang menjelaskan data login. Input ini termasuk perangkat, browser, dan input geolokasi. Amazon Fraud Detector juga menggunakan data login yang Anda berikan untuk terus menghitung variabel agregat yang menggambarkan perilaku pengguna sebelumnya. Contoh perilaku pengguna termasuk berapa kali pengguna masuk dari alamat IP tertentu. Menggunakan pengayaan dan agregat tambahan ini, Amazon Fraud Detector dapat menghasilkan kinerja model yang kuat dari sekumpulan kecil input dari peristiwa login Anda.

Model Account Takeover Insights mendeteksi contoh di mana akun yang sah diakses oleh aktor jahat, terlepas dari apakah aktor jahat itu manusia atau robot. Model ini menghasilkan skor tunggal yang menunjukkan risiko relatif kompromi akun. Akun yang mungkin telah disusupi ditandai sebagai akun berisiko tinggi. Anda dapat memproses akun berisiko tinggi dengan salah satu dari dua cara. Anda juga dapat menerapkan verifikasi identitas tambahan. Atau, Anda dapat mengirim akun ke antrian untuk penyelidikan manual.

Memilih sumber data

Model Account Takeover Insights dilatih pada kumpulan data yang disimpan secara internal, di Amazon Fraud Detector. Untuk menyimpan data peristiwa login Anda dengan Amazon Fraud Detector, buat file CSV dengan peristiwa login pengguna. Untuk setiap acara, sertakan data login seperti stempel waktu acara, ID pengguna, alamat IP, agen pengguna, dan apakah data login valid. Setelah membuat file CSV, pertama upload file ke Amazon Fraud Detector, dan kemudian gunakan fitur impor untuk menyimpan data. Anda kemudian dapat melatih model Anda menggunakan data yang disimpan. Untuk informasi selengkapnya tentang menyimpan kumpulan data acara Anda dengan Amazon Fraud Detector, lihat Simpan data acara Anda secara internal dengan Amazon Fraud Detector

Mempersiapkan data

Amazon Fraud Detector mengharuskan Anda memberikan data login akun pengguna Anda dalam file nilai yang dipisahkan koma (CSV) yang dikodekan dalam format UTF-8. Baris pertama file CSV Anda harus berisi header file. Header file terdiri dari metadata peristiwa dan variabel peristiwa yang

menggambarkan setiap elemen data. Data acara mengikuti header. Setiap baris dalam data acara terdiri dari data dari satu peristiwa login.

Untuk model Wawasan Pengambilalihan Akun, Anda harus menyediakan metadata peristiwa dan variabel peristiwa berikut di baris header file CSV Anda.

Metadata acara

Kami menyarankan Anda memberikan metadata berikut di header file CSV Anda. Metadata acara harus dalam huruf besar.

- EVENT_ID - Sebuah identifier unik untuk acara login.
- ENTITY_TYPE - Entitas yang melakukan peristiwa login, seperti pedagang atau pelanggan.
- ENTITY_ID - Sebuah identifier untuk entitas melakukan peristiwa login.
- EVENT_TIMESTAMP - Stempel waktu saat peristiwa login terjadi. Stempel waktu harus dalam standar ISO 8601 di UTC.
- EVENT_LABEL (direkomendasikan) - Label yang mengklasifikasikan acara sebagai penipuan atau sah. Anda dapat menggunakan label apa pun, seperti “penipuan”, “legit”, “1", atau “0".

> **Note**
>
> - Metadata peristiwa harus dalam huruf besar. Ini peka huruf besar/kecil.
> - Label tidak diperlukan untuk acara login. Namun, kami menyarankan Anda menyertakan metadata EVENT_LABEL dan memberikan label untuk peristiwa login Anda. Tidak apa-apa jika labelnya tidak lengkap atau sporadis. Jika Anda memberikan label, Amazon Fraud Detector akan menggunakannya untuk menghitung Account Takeover Discovery Rate secara otomatis dan menampilkannya dalam bagan dan tabel kinerja model.

Variabel peristiwa

Untuk model Account Takeover Insights, ada variabel wajib (wajib) yang harus Anda sediakan dan variabel opsional. Ketika Anda membuat variabel Anda, pastikan untuk menetapkan variabel ke jenis variabel yang tepat. Sebagai bagian dari proses pelatihan model, Amazon Fraud Detector menggunakan tipe variabel yang terkait dengan variabel untuk melakukan pengayaan variabel dan rekayasa fitur.

> **Note**
>
> Nama variabel peristiwa harus dalam huruf kecil. Mereka peka huruf besar/kecil.

Variabel wajib

Variabel berikut diperlukan untuk melatih model Accounts Takeover Insights.

Kategori	Jenis variabel	Deskripsi
Alamat IP	IP_ALAMAT	Alamat IP yang digunakan dalam acara login
Browser dan perangkat	AGEN PENGGUNA	Browser, perangkat, dan OS yang digunakan dalam acara login
Kredensi yang valid	VALIDCRED	Menunjukkan apakah kredensyal yang digunakan untuk login valid

Variabel opsional

Variabel berikut bersifat opsional untuk melatih model Accounts Takeover Insights.

Kategori	Tipe	Deskripsi
Browser dan perangkat	SIDIK JARI	Pengidentifikasi unik untuk browser atau sidik jari perangkat
Id Sesi	SESSION_ID	Pengidentifikasi untuk sesi otentikasi
Label	EVENT_LABEL	Label yang mengklasifikasikan acara sebagai penipuan atau sah. Anda dapat menggunak

Kategori	Tipe	Deskripsi
		an label apa pun, seperti “penipuan”, “legit”, “1", atau “0".
Stempel Waktu	LABEL_TIMESTAMP	Stempel waktu saat label terakhir diperbarui. Ini diperlukan jika EVENT_LABEL disediakan.

> **Note**
>
> - Anda dapat memberikan nama variabel untuk kedua variabel wajib variabel opsional. Sangat penting bahwa setiap variabel wajib dan opsional ditetapkan ke jenis variabel yang tepat.
> - Anda dapat memberikan variabel tambahan. Namun, Amazon Fraud Detector tidak akan menyertakan variabel ini untuk melatih model Accounts Takeover Insights.

Memilih data

Mengumpulkan data merupakan langkah penting untuk membuat model Account Takeover Insights Anda. Saat Anda mulai mengumpulkan data login Anda, pertimbangkan persyaratan dan rekomendasi berikut:

Diperlukan

- Berikan setidaknya 1.500 contoh akun pengguna, masing-masing dengan setidaknya dua peristiwa login terkait.
- Dataset Anda harus mencakup setidaknya 30 hari peristiwa login. Anda nantinya dapat menentukan rentang waktu tertentu dari peristiwa yang akan digunakan untuk melatih model.

Direkomendasikan

- Dataset Anda mencakup contoh peristiwa login yang gagal. Anda dapat secara opsional memberi label login yang gagal ini sebagai “penipuan” atau “sah.”

- Siapkan data historis dengan acara login yang mencakup lebih dari enam bulan dan sertakan 100 ribu entitas.

Jika Anda tidak memiliki kumpulan data yang sudah memenuhi persyaratan minimum, pertimbangkan streaming data peristiwa ke Amazon Fraud Detector dengan memanggil operasi SendEventAPI.

Memvalidasi data

Sebelum membuat model Account Takeover Insights, Amazon Fraud Detector memeriksa apakah metadata dan variabel yang Anda sertakan dalam kumpulan data untuk melatih model memenuhi persyaratan ukuran dan format. Untuk informasi selengkapnya, lihat Validasi data. Ini juga memeriksa persyaratan lain. Jika dataset tidak lulus validasi, model tidak dibuat. Agar model berhasil dibuat, pastikan untuk memperbaiki data yang tidak lulus validasi sebelum Anda berlatih lagi.

Kesalahan kumpulan data umum

Saat memvalidasi kumpulan data untuk melatih model Account Takeover Insights, Amazon Fraud Detector memindai masalah ini dan masalah lainnya dan menimbulkan error jika mengalami satu atau beberapa masalah.

- File CSV tidak dalam format UTF-8.
- Header file CSV tidak berisi setidaknya satu dari metadata berikut:`EVENT_ID`,, `ENTITY_ID` atau. `EVENT_TIMESTAMP`
- Header file CSV tidak berisi setidaknya satu variabel dari jenis variabel berikut:`IP_ADDRESS`,`USERAGENT`, atau`VALIDCRED`.
- Ada lebih dari satu variabel yang terkait dengan tipe variabel yang sama.
- Lebih dari 0,1% nilai dalam `EVENT_TIMESTAMP` berisi nol atau nilai selain format tanggal dan stempel waktu yang didukung.
- Jumlah hari antara acara pertama dan terakhir kurang dari 30 hari.
- Lebih dari 10% variabel tipe `IP_ADDRESS` variabel tidak valid atau null.
- Lebih dari 50% variabel tipe `USERAGENT` variabel mengandung null.
- Semua variabel dari tipe `VALIDCRED` variabel diatur `kefalse`.

Membangun model

Model Amazon Fraud Detector belajar mendeteksi penipuan untuk jenis peristiwa tertentu. Di Amazon Fraud Detector, pertama-tama Anda membuat model, yang berfungsi sebagai wadah untuk versi model Anda. Setiap kali Anda melatih model, versi baru dibuat. Untuk detail tentang cara membuat dan melatih model menggunakan AWS Konsol, lihatLangkah 3: Membuat Model.

Setiap model memiliki variabel skor model yang sesuai. Amazon Fraud Detector membuat variabel ini atas nama Anda saat Anda membuat model. Anda dapat menggunakan variabel ini dalam ekspresi aturan Anda untuk menafsirkan skor model Anda selama evaluasi penipuan.

Latih dan terapkan model menggunakan AWS SDK for Python (Boto3)

Versi model dibuat dengan memanggil `CreateModel` dan `CreateModelVersion` operasi. `CreateModel`memulai model, yang bertindak sebagai wadah untuk versi model Anda. `CreateModelVersion`memulai proses pelatihan, yang menghasilkan versi model tertentu. Versi baru dari solusi dibuat setiap kali Anda meneleponCreateModelVersion.

Contoh berikut menunjukkan permintaan sampel untuk `CreateModel` API. Contoh ini membuat jenis model Wawasan Penipuan Online dan mengasumsikan Anda telah membuat jenis acara. `sample_registration` Untuk detail tambahan tentang membuat jenis acara, lihatMembuat jenis acara.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_model (
modelId = 'sample_fraud_detection_model',
eventTypeName = 'sample_registration',
modelType = 'ONLINE_FRAUD_INSIGHTS')
```

Latih versi pertama Anda menggunakan CreateModelVersionAPI. Untuk `TrainingDataSource` dan `ExternalEventsDetail` tentukan sumber dan lokasi Amazon S3 dari kumpulan data pelatihan. Untuk `TrainingDataSchema` menentukan bagaimana Amazon Fraud Detector harus menafsirkan data pelatihan, khususnya variabel peristiwa yang akan disertakan dan cara mengklasifikasikan label peristiwa. Secara default, Amazon Fraud Detector mengabaikan peristiwa yang tidak berlabel. Kode contoh ini digunakan `AUTO` `unlabeledEventsTreatment` untuk menentukan bahwa Amazon Fraud Detector memutuskan cara menggunakan peristiwa yang tidak berlabel.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_model_version (
modelId = 'sample_fraud_detection_model',
modelType = 'ONLINE_FRAUD_INSIGHTS',
trainingDataSource = 'EXTERNAL_EVENTS',
trainingDataSchema = {
    'modelVariables' : ['ip_address', 'email_address'],
    'labelSchema' : {
        'labelMapper' : {
            'FRAUD' : ['fraud'],
            'LEGIT' : ['legit']
        }
       unlabeledEventsTreatment = 'AUTO'
    }
},
externalEventsDetail = {
    'dataLocation' : 's3://bucket/file.csv',
    'dataAccessRoleArn' : 'role_arn'
}
)
```

Permintaan yang berhasil akan menghasilkan versi model baru dengan statusTRAINING_IN_PROGRESS. Kapan saja selama pelatihan, Anda dapat membatalkan pelatihan dengan menelepon UpdateModelVersionStatus dan memperbarui status keTRAINING_CANCELLED. Setelah pelatihan selesai, status versi model akan diperbarui keTRAINING_COMPLETE. Anda dapat meninjau performa model menggunakan konsol Amazon Fraud Detector atau dengan meneleponDescribeModelVersions. Untuk informasi lebih lanjut tentang cara menafsirkan skor dan kinerja model, lihat Skor model danMetrik kinerja model.

Setelah meninjau kinerja model, aktifkan model agar tersedia untuk digunakan oleh Detektor dalam prediksi penipuan waktu nyata. Amazon Fraud Detector akan menerapkan model di beberapa zona ketersediaan untuk redundansi dengan auto-scaling diaktifkan untuk memastikan model menskalakan dengan jumlah prediksi penipuan yang Anda buat. Untuk mengaktifkan model, panggil UpdateModelVersionStatus API dan perbarui statusnya keACTIVE.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_model_version_status (
```

```
modelId = 'sample_fraud_detection_model',
modelType = 'ONLINE_FRAUD_INSIGHTS',
modelVersionNumber = '1.00',
status = 'ACTIVE'
)
```

Skor model

Amazon Fraud Detector menghasilkan skor model yang berbeda untuk jenis model yang berbeda.

Untuk model Account Takeover Insights (ATI), Amazon Fraud Detector hanya menggunakan nilai agregat (nilai yang dihitung dengan menggabungkan sekumpulan variabel mentah) untuk menghasilkan skor model. Skor -1 dihasilkan untuk peristiwa pertama entitas baru, menunjukkan risiko yang tidak diketahui. Ini karena untuk entitas baru, nilai yang digunakan untuk menghitung agregat akan menjadi nol atau nol. Model Account Takeeover Insights (ATI) menghasilkan skor model antara 0 dan 1000 untuk semua peristiwa berikutnya untuk entitas yang sama dan untuk entitas yang ada, di mana 0 menunjukkan risiko penipuan rendah dan 1000 menunjukkan risiko penipuan yang tinggi. Untuk model ATI, skor model terkait langsung dengan tingkat tantangan (CR). Misalnya, skor 500 sesuai dengan perkiraan tingkat tantangan 5% sedangkan skor 900 sesuai dengan perkiraan tingkat tantangan 0,1%.

Untuk model Online Fraud Insights (OFI) dan Transaction Fraud Insights (TFI), Amazon Fraud Detector menggunakan nilai agregat (nilai yang dihitung dengan menggabungkan satu set variabel mentah) dan nilai mentah (nilai yang disediakan untuk variabel) untuk menghasilkan skor model. Skor model bisa antara 0 dan 1000, di mana 0 menunjukkan risiko penipuan rendah dan 1000 menunjukkan risiko penipuan yang tinggi. Untuk model OFI dan TFI, skor model berhubungan langsung dengan tingkat positif palsu (FPR). Misalnya, skor 600 sesuai dengan perkiraan 10% tingkat positif palsu sedangkan skor 900 sesuai dengan perkiraan 2% tingkat positif palsu. Tabel berikut memberikan rincian tentang bagaimana skor model tertentu berkorelasi dengan perkiraan tingkat positif palsu.

Skor model	Perkiraan FPR
975	0,50%
950	1%
900	2%

Skor model	Perkiraan FPR
860	3%
775	5%
700	7%
600	10%

Metrik kinerja model

Setelah pelatihan model selesai, Amazon Fraud Detector memvalidasi kinerja model menggunakan 15% data Anda yang tidak digunakan untuk melatih model. Anda dapat mengharapkan model Amazon Fraud Detector terlatih Anda memiliki kinerja deteksi penipuan dunia nyata yang mirip dengan metrik kinerja validasi.

Sebagai bisnis, Anda harus menyeimbangkan antara mendeteksi lebih banyak penipuan, dan menambahkan lebih banyak gesekan ke pelanggan yang sah. Untuk membantu memilih keseimbangan yang tepat, Amazon Fraud Detector menyediakan alat berikut untuk menilai kinerja model:

- Bagan distribusi skor — Histogram distribusi skor model mengasumsikan contoh populasi 100.000 peristiwa. Sumbu Y kiri mewakili peristiwa yang sah dan sumbu Y kanan mewakili peristiwa penipuan. Anda dapat memilih ambang model tertentu dengan mengklik area bagan. Ini akan memperbarui tampilan yang sesuai dalam matriks kebingungan dan bagan ROC.
- Matriks kebingungan — Merangkum akurasi model untuk ambang skor tertentu dengan membandingkan prediksi model versus hasil aktual. Amazon Fraud Detector mengasumsikan contoh populasi 100.000 peristiwa. Distribusi penipuan dan peristiwa yang sah mensimulasikan tingkat penipuan dalam bisnis Anda.
  - Positif sejati — Model memprediksi penipuan dan acara tersebut sebenarnya penipuan.
  - Positif palsu — Model memprediksi penipuan tetapi acara tersebut sebenarnya sah.
  - Negatif sejati — Model memprediksi sah dan acara tersebut sebenarnya sah.
  - Negatif palsu — Model memprediksi sah tetapi acara tersebut sebenarnya penipuan.
  - True positive rate (TPR) — Persentase total penipuan yang dideteksi model. Juga dikenal sebagai capture rate.

  - Tingkat positif palsu (FPR) — Persentase dari total peristiwa sah yang salah diprediksi sebagai penipuan.
- Receiver Operator Curve (ROC) — Memplot laju positif sebenarnya sebagai fungsi dari laju positif palsu di atas semua ambang batas skor model yang mungkin. Lihat bagan ini dengan memilih Metrik Lanjutan.
- Area di bawah kurva (AUC) — Merangkum TPR dan FPR di semua ambang batas skor model yang mungkin. Model tanpa daya prediksi memiliki AUC 0,5, sedangkan model sempurna memiliki skor 1,0.
- Rentang ketidakpastian — Ini menunjukkan kisaran AUC yang diharapkan dari model. Rentang yang lebih besar (perbedaan batas atas dan bawah AUC > 0.1) berarti ketidakpastian model yang lebih tinggi. Jika rentang ketidakpastian besar (>0.1), pertimbangkan untuk menyediakan lebih banyak peristiwa berlabel dan latih kembali modelnya.

Untuk menggunakan metrik kinerja model

1. Mulailah dengan bagan distribusi Skor untuk meninjau distribusi skor model untuk penipuan dan peristiwa yang sah. Idealnya, Anda akan memiliki pemisahan yang jelas antara penipuan dan peristiwa yang sah. Ini menunjukkan model dapat secara akurat mengidentifikasi peristiwa mana yang curang dan mana yang sah. Pilih ambang model dengan mengklik area bagan. Anda dapat melihat bagaimana menyesuaikan ambang skor model memengaruhi tingkat positif positif dan positif palsu Anda yang sebenarnya.

   > **Note**
   >
   > Bagan distribusi skor memplot penipuan dan peristiwa yang sah pada dua sumbu Y yang berbeda. Sumbu Y kiri mewakili peristiwa yang sah dan sumbu Y kanan mewakili peristiwa penipuan.

2. Tinjau matriks Kebingungan. Bergantung pada ambang skor model yang Anda pilih, Anda dapat melihat dampak simulasi berdasarkan sampel 100.000 peristiwa. Distribusi penipuan dan peristiwa yang sah mensimulasikan tingkat penipuan dalam bisnis Anda. Gunakan informasi ini untuk menemukan keseimbangan yang tepat antara tingkat positif sejati dan tingkat positif palsu.
3. Untuk detail tambahan, pilih Metrik Lanjutan. Gunakan bagan ROC untuk memahami hubungan antara tingkat positif sejati dan tingkat positif palsu untuk ambang skor model apa pun. Kurva ROC dapat membantu Anda menyempurnakan tradeoff antara tingkat positif sejati dan tingkat positif palsu.

> **Note**
>
> Anda juga dapat meninjau metrik dalam bentuk tabel dengan memilih Tabel. Tampilan tabel juga menunjukkan Presisi metrik. Presisi adalah persentase kejadian penipuan yang diprediksi dengan benar sebagai penipuan dibandingkan dengan semua peristiwa yang diprediksi sebagai penipuan.

4. Gunakan metrik kinerja untuk menentukan ambang model optimal untuk bisnis Anda berdasarkan sasaran dan kasus penggunaan deteksi penipuan Anda. Misalnya, jika Anda berencana menggunakan model untuk mengklasifikasikan pendaftaran akun baru sebagai risiko tinggi, sedang, atau rendah, Anda perlu mengidentifikasi dua skor ambang sehingga Anda dapat menyusun tiga ketentuan aturan sebagai berikut:

   - Skor > X berisiko tinggi
   - Skor < X but > Y adalah risiko sedang
   - Skor < Y berisiko rendah

Pentingnya variabel model

Kepentingan variabel model adalah fitur Amazon Fraud Detector yang memberi peringkat variabel model dalam versi model. Setiap variabel model diberikan nilai berdasarkan kepentingan relatifnya terhadap kinerja keseluruhan model Anda. Variabel model dengan nilai tertinggi lebih penting bagi model daripada variabel model lain dalam kumpulan data untuk versi model tersebut, dan terdaftar di bagian atas secara default. Demikian juga, variabel model dengan nilai terendah terdaftar di bagian bawah secara default dan paling tidak penting dibandingkan dengan variabel model lainnya. Dengan menggunakan nilai kepentingan variabel model, Anda dapat memperoleh wawasan tentang input apa yang mendorong kinerja model Anda.

Anda dapat melihat nilai kepentingan variabel model untuk versi model terlatih di konsol Amazon Fraud Detector atau dengan menggunakan DescribeModelVersionAPI.

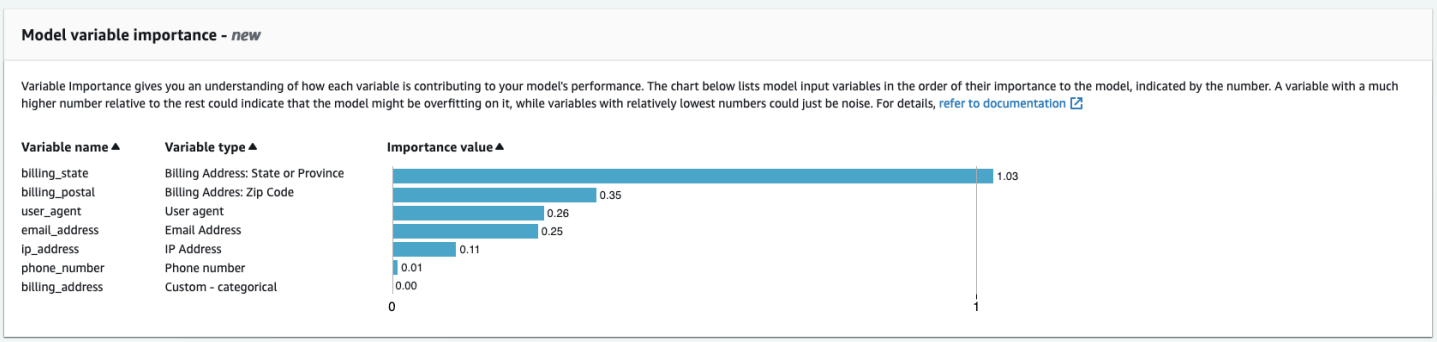
Kepentingan variabel model memberikan serangkaian nilai berikut untuk setiap Variabel yang digunakan untuk melatih Versi Model.

- Jenis Variabel: Jenis variabel (misalnya, alamat IP atau Email). Untuk informasi selengkapnya, lihat Jenis variabel. Untuk model Account Takeover Insights (ATI), Amazon Fraud Detector memberikan nilai kepentingan variabel untuk tipe variabel mentah dan agregat. Jenis variabel

mentah ditugaskan ke variabel yang Anda berikan. Jenis variabel agregat ditetapkan ke satu set variabel mentah yang telah digabungkan oleh Amazon Fraud Detector untuk menghitung nilai kepentingan agregat.

- Nama Variabel: Nama variabel peristiwa yang digunakan untuk melatih versi model (misalnya,`ip_address`,`email_address`,`are_creadentials_valid`). Untuk tipe variabel agregat, nama semua variabel yang digunakan untuk menghitung nilai kepentingan variabel agregat dicantumkan.
- Nilai Pentingnya Variabel: Angka yang mewakili kepentingan relatif dari variabel mentah atau agregat terhadap kinerja model. Kisaran tipikal: 0—10

Di konsol Amazon Fraud Detector, nilai kepentingan variabel model ditampilkan sebagai berikut untuk model Online Fraud Insights (OFI) atau Transaction Fraud Insights (TFI). Model Account Takeover Insight (ATI) akan memberikan nilai kepentingan variabel agregat selain nilai kepentingan variabel mentah. Bagan visual memudahkan untuk melihat kepentingan relatif antara variabel dengan garis putus-putus vertikal yang memberikan referensi ke nilai kepentingan variabel peringkat tertinggi.



Amazon Fraud Detector menghasilkan nilai kepentingan variabel untuk setiap versi model Fraud Detector tanpa biaya tambahan.

> **Important**
>
> Versi model yang dibuat sebelum 9 Juli 2021 tidak memiliki nilai kepentingan variabel. Anda harus melatih versi baru model Anda untuk menghasilkan nilai kepentingan variabel model.

Menggunakan nilai kepentingan variabel model

Anda dapat menggunakan nilai kepentingan variabel model untuk mendapatkan wawasan tentang apa yang mendorong kinerja model Anda naik atau turun dan variabel mana yang paling berkontribusi. Dan kemudian tweak model Anda untuk meningkatkan kinerja secara keseluruhan.

Lebih khusus lagi, untuk meningkatkan kinerja model Anda, periksa nilai kepentingan variabel terhadap pengetahuan domain Anda dan masalah debug dalam data pelatihan. Misalnya, jika ID Akun digunakan sebagai masukan ke model dan terdaftar di bagian atas, lihat nilai kepentingan variabelnya. Jika nilai kepentingan variabel secara signifikan lebih tinggi daripada nilai lainnya, maka model Anda mungkin terlalu sesuai dengan pola penipuan tertentu (misalnya, semua peristiwa penipuan berasal dari ID Akun yang sama). Namun, mungkin juga terjadi kebocoran label jika variabel tergantung pada label penipuan. Bergantung pada hasil analisis Anda berdasarkan pengetahuan domain Anda, Anda mungkin ingin menghapus variabel dan melatih dengan kumpulan data yang lebih beragam, atau mempertahankan model apa adanya.

Demikian pula, lihat variabel peringkat terakhir. Jika nilai kepentingan variabel secara signifikan lebih rendah daripada nilai lainnya, maka variabel model ini mungkin tidak penting dalam melatih model Anda. Anda dapat mempertimbangkan untuk menghapus variabel untuk melatih versi model yang lebih sederhana. Jika model Anda memiliki beberapa variabel, seperti hanya dua variabel, Amazon Fraud Detector masih memberikan nilai kepentingan variabel dan memberi peringkat variabel. Namun, wawasan dalam hal ini akan terbatas.

Important

1. Jika Anda melihat variabel yang hilang dalam bagan kepentingan variabel Model, itu mungkin karena salah satu alasan berikut. Pertimbangkan untuk memodifikasi variabel dalam kumpulan data Anda dan latih kembali model Anda.
   - Hitungan nilai unik untuk variabel dalam kumpulan data pelatihan lebih rendah dari 100.
   - Lebih besar dari 0,9 nilai untuk variabel hilang dari kumpulan data pelatihan.
2. Anda perlu melatih versi model baru setiap kali Anda ingin menyesuaikan variabel input model Anda.

Mengevaluasi nilai kepentingan variabel model

Kami menyarankan Anda mempertimbangkan hal berikut ketika Anda mengevaluasi nilai kepentingan variabel model:

- Nilai kepentingan variabel harus selalu dievaluasi dalam kombinasi dengan pengetahuan domain.
- Periksa nilai kepentingan variabel dari variabel relatif terhadap nilai kepentingan variabel dari variabel lain dalam versi model. Jangan mempertimbangkan nilai kepentingan variabel untuk satu variabel secara independen.
- Bandingkan nilai kepentingan variabel dari variabel dalam versi model yang sama. Jangan membandingkan nilai kepentingan variabel dari variabel yang sama di seluruh versi model karena nilai kepentingan variabel variabel dalam versi model mungkin berbeda dari nilai variabel yang sama dalam versi model yang berbeda. Jika Anda menggunakan variabel dan kumpulan data yang sama untuk melatih versi model yang berbeda, ini tidak selalu menghasilkan nilai kepentingan variabel yang sama.

Melihat peringkat kepentingan variabel model

Setelah pelatihan model selesai, Anda dapat melihat peringkat kepentingan variabel model dari versi model terlatih Anda di konsol Amazon Fraud Detector atau dengan menggunakan DescribeModelVersionAPI.

Untuk melihat peringkat kepentingan variabel model menggunakan konsol,

1. Buka AWS Konsol dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Model.
3. Pilih model Anda dan kemudian versi model Anda.
4. Pastikan bahwa tab Ikhtisar dipilih.
5. Gulir ke bawah untuk melihat panel kepentingan variabel Model.

Memahami bagaimana nilai kepentingan variabel model dihitung

Setelah menyelesaikan setiap pelatihan versi model, Amazon Fraud Detector secara otomatis menghasilkan nilai kepentingan variabel model dan metrik kinerja model. Untuk ini, Amazon Fraud Detector menggunakan Shapley Additive Explanations (SHAP). SHAP pada dasarnya adalah

kontribusi rata-rata yang diharapkan dari variabel model setelah semua kemungkinan kombinasi dari semua variabel model telah dipertimbangkan.

SHAP pertama-tama memberikan kontribusi dari setiap variabel model untuk prediksi suatu peristiwa. Kemudian, ia mengumpulkan prediksi ini untuk membuat peringkat variabel di tingkat model. Untuk menetapkan kontribusi dari setiap variabel model untuk prediksi, SHAP mempertimbangkan perbedaan dalam output model di antara semua kemungkinan kombinasi variabel. Dengan memasukkan semua kemungkinan termasuk atau menghapus set variabel tertentu untuk menghasilkan output model, SHAP dapat secara akurat mengakses pentingnya setiap variabel model. Ini sangat penting ketika variabel model sangat berkorelasi satu sama lain.

Model ML, dalam banyak kasus, tidak memungkinkan Anda untuk menghapus variabel. Sebagai gantinya, Anda dapat mengganti variabel yang dihapus atau hilang dalam model dengan nilai variabel yang sesuai dari satu atau lebih garis dasar (misalnya, peristiwa non-penipuan). Memilih instance dasar yang tepat bisa jadi sulit, tetapi Amazon Fraud Detector mempermudah hal ini dengan menetapkan baseline ini sebagai rata-rata populasi untuk Anda.

Impor SageMaker model

Anda dapat secara opsional mengimpor model yang SageMaker di-host ke Amazon Fraud Detector. Mirip dengan SageMaker model, model dapat ditambahkan ke detektor dan menghasilkan prediksi penipuan menggunakan API. `GetEventPrediction` Sebagai bagian dari `GetEventPrediction` permintaan, Amazon Fraud Detector akan memanggil SageMaker titik akhir Anda dan meneruskan hasilnya ke aturan Anda.

Anda dapat mengonfigurasi Amazon Fraud Detector untuk menggunakan variabel peristiwa yang dikirim sebagai bagian dari `GetEventPrediction` permintaan. Jika Anda memilih untuk menggunakan variabel acara, Anda harus memberikan template masukan. Amazon Fraud Detector akan menggunakan template ini untuk mengubah variabel event Anda menjadi payload input yang diperlukan untuk memanggil endpoint. SageMaker Atau, Anda dapat mengonfigurasi SageMaker model Anda untuk menggunakan ByteBuffer yang dikirim sebagai bagian dari permintaan. `GetEventPrediction`

Amazon Fraud Detector mendukung pengimporan SageMaker algoritme yang menggunakan format input JSON atau CSV dan format keluaran JSON atau CSV. Contoh SageMaker algoritma yang didukung termasuk XGBoost, Linear Learner, dan Random Cut Forest.

Impor SageMaker model menggunakan AWS SDK for Python (Boto3)

Untuk mengimpor SageMaker model, gunakan `PutExternalModel` API. Contoh berikut mengasumsikan SageMaker titik akhir `sagemaker-transaction-model` telah digunakan, adalah `InService` status, dan menggunakan algoritma XGBoost.

Konfigurasi input menentukan yang akan menggunakan variabel peristiwa untuk membangun input model (`useEventVariables`diatur keTRUE). Format input adalah TEXT_CSV, mengingat XGBoost membutuhkan input CSV. csvInputTemplate Menentukan bagaimana membangun input CSV dari variabel yang dikirim sebagai bagian dari permintaan. `GetEventPrediction` Contoh ini mengasumsikan Anda telah membuat variabel`order_amt`,`prev_amt`, `hist_amt` dan`payment_type`.

Konfigurasi keluaran menentukan format respons SageMaker model, dan memetakan indeks CSV yang sesuai ke variabel Amazon Fraud Detector. `sagemaker_output_score` Setelah dikonfigurasi, Anda dapat menggunakan variabel output dalam aturan.

> **Note**
>
> Output dari SageMaker model harus dipetakan ke variabel dengan sumber`EXTERNAL_MODEL_SCORE`. Anda tidak dapat membuat variabel ini di konsol menggunakan Variabel. Sebagai gantinya, Anda harus membuatnya saat mengonfigurasi impor model Anda.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_external_model (
modelSource = 'SAGEMAKER',
modelEndpoint = 'sagemaker-transaction-model',
invokeModelEndpointRoleArn = 'your_SagemakerExecutionRole_arn',
inputConfiguration = {
    'useEventVariables' : True,
    'eventTypeName' : 'sample_transaction',
    'format' : 'TEXT_CSV',
    'csvInputTemplate' : '{{order_amt}}, {{prev_amt}}, {{hist_amt}}, {{payment_type}}'
},

outputConfiguration = {
```

```
    'format' : 'TEXT_CSV',
    'csvIndexToVariableMap' : {
        '0' : 'sagemaker_output_score'
    }
},

modelEndpointStatus = 'ASSOCIATED'
)
```

Menghapus versi model atau model

Anda dapat menghapus model dan versi model di Amazon Fraud Detector, asalkan tidak terkait dengan versi detektor. Saat Anda menghapus model, Amazon Fraud Detector menghapus model tersebut secara permanen dan data tidak lagi disimpan di Amazon Fraud Detector.

Anda juga dapat menghapus SageMaker model Amazon jika tidak terkait dengan versi detektor. Menghapus SageMaker model memutuskannya dari Amazon Fraud Detector, tetapi modelnya tetap tersedia SageMaker.

Untuk menghapus versi model

Anda hanya dapat menghapus versi model yang ada dalam`Ready to deploy` status. Untuk mengubah versi model dari`ACTIVE``Ready to deploy` status, undeploy versi model.

1. Masuk keAWS Management Console dan buka konsol Amazon Fraud Detector di https://console.aws.amazon.com/frauddetector.
2. Pada panel navigasi kiri konsol Amazon Fraud Detector, pilih Model.
3. Pilih model yang ingin yang ingin hapus.
4. Pilih versi model yang ingin ingin hapus.
5. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
6. Masukkan nama versi model, lalu pilih Hapus versi model.

Untuk undeploy versi model

Anda tidak dapat membuka versi model yang digunakan oleh versi detektor apa pun (`ACTIVE`,`INACTIVE`,`DRAFT`). Oleh karena itu, untuk membuka versi model yang digunakan oleh versi detektor, pertama-tama hapus versi model dari versi detektor.

1. Pada panel navigasi kiri konsol Amazon Fraud Detector, pilih Model.
2. Pilih model yang berisi versi model yang ingin Anda undeploy.
3. Pilih versi model yang ingin ingin hapus.
4. Pilih Tindakan, lalu pilih Undeploy versi model.

Untuk menghapus model

Sebelum menghapus model, Anda harus terlebih dahulu menghapus semua versi model dan dikaitkan dengan model.

1. Pada panel navigasi kiri konsol Amazon Fraud Detector, pilih Model.
2. Pilih model yang ingin ingin hapus.
3. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
4. Masukkan nama model, lalu pilih Hapus model.

Untuk menghapus SageMaker model Amazon

1. Pada panel navigasi kiri konsol Amazon Fraud Detector, pilih Model.
2. Pilih SageMaker model yang ingin ingin hapus.
3. Pilih Tindakan, dan kemudian pilih Hapus model.
4. Masukkan nama model dan kemudian pilih Hapus SageMaker model.

Detektor

Detektor adalah wadah yang berisi logika deteksi penipuan, seperti model dan aturan, untuk satu peristiwa bisnis tertentu yang ingin Anda evaluasi untuk penipuan. Pertama-tama Anda membuat detektor dengan menentukan peristiwa yang telah Anda tentukan dan secara opsional menambahkan versi model yang sudah dibuat dan dilatih oleh Amazon Fraud Detector untuk acara tersebut.

Anda kemudian menambahkan aturan dan perintah eksekusi aturan ke detektor untuk membuat versi detektor. Versi detektor mendefinisikan aturan dan secara opsional model yang akan dijalankan sebagai bagian dari permintaan untuk menghasilkan prediksi penipuan. Anda dapat menambahkan aturan apa pun yang ditentukan dalam detektor ke versi detektor. Anda juga dapat menambahkan model apa pun yang dilatih pada jenis peristiwa yang dievaluasi ke versi detektor. Detektor dapat memiliki beberapa versi, dengan setiap versi memiliki aturan dan perintah eksekusi aturan yang berbeda untuk memenuhi beberapa kasus penggunaan.

Setiap versi detektor harus memiliki status`DRAFT`,`ACTIVE`, atau`INACTIVE`. Hanya satu versi detektor yang bisa masuk`ACTIVE`status pada suatu waktu. Amazon Fraud Detector menggunakan versi detektor dengan`ACTIVE`status untuk menghasilkan prediksi penipuan.

Buat detektor

Anda membuat detektor dengan menentukan jenis peristiwa yang telah Anda tetapkan. Anda dapat menambahkan model yang sudah dilatih dan diterapkan oleh Amazon Fraud Detector. Jika Anda menambahkan model, Anda dapat menggunakan skor model yang dihasilkan oleh Amazon Fraud Detector dalam ekspresi aturan Anda saat membuat aturan (misalnya,`$model score < 90`).

Anda dapat membuat detektor di konsol Amazon Fraud Detector, menggunakanPutDetectorAPI, menggunakanmenempatkan detektorperintah, atau menggunakanAWSSDK. Jika Anda menggunakan API, perintah, atau SDK untuk membuat detektor, setelah Anda membuat detektor ikuti instruksiBuat versi detektor.

Membuat detektor di konsol Amazon Fraud Detector

Contoh ini mengasumsikan bahwa Anda telah membuat jenis peristiwa dan juga telah membuat dan menerapkan versi model yang ingin Anda gunakan untuk prediksi penipuan.

Langkah 1: Bangun detektor

1. Di panel navigasi kiri konsol Amazon Fraud Detector, pilihDetektor.

2. PilihBuat detektor.
3. DalamTentukan detail detektorhalaman, masukkan`sample_detector`untuk nama detektor. Secara opsional, masukkan deskripsi untuk detektor, sepertimy `sample fraud detector`.
4. UntukJenis Acara, pilih jenis peristiwa yang telah Anda buat untuk prediksi penipuan.
5. Pilih Selanjutnya.

Langkah 2: Tambahkan versi model yang diterapkan

1. Perhatikan bahwa ini adalah langkah opsional. Anda tidak perlu menambahkan model ke detektor Anda. Untuk melewati langkah ini, pilihBerikutnya.
2. DalamTambahkan model - opsional, pilihTambahkan Model.
3. DalamTambahkan modelhalaman, untukPilih model, pilih nama model Amazon Fraud Detector yang Anda gunakan sebelumnya. UntukPilih versi, pilih versi model model yang dikerahkan.
4. PilihTambahkan model.
5. Pilih Selanjutnya.

Langkah 3: Tambahkan aturan

Aturan adalah kondisi yang memberi tahu Amazon Fraud Detector cara menafsirkan nilai variabel saat mengevaluasi prediksi penipuan. Contoh ini akan membuat tiga aturan menggunakan skor model sebagai nilai variabel:`high_fraud_risk`,`medium_fraud_risk`, dan`low_fraud_risk`. Untuk membuat aturan, ekspresi aturan, urutan eksekusi aturan, dan hasil Anda sendiri, gunakan nilai yang sesuai untuk model dan kasus penggunaan Anda.

1. DalamTambahkan aturanhalaman, di bawahTentukan aturan, masuklah`high_fraud_risk`untuk nama aturan dan di bawahDeskripsi - opsional, masuklah**`This rule captures events with a high ML model score`**sebagai deskripsi untuk aturan.
2. DalamEkspresi, masukkan ekspresi aturan berikut menggunakan bahasa ekspresi aturan sederhana Amazon Fraud Detector:

   ```
   $sample_fraud_detection_model_insightscore > 900
   ```

3. DalamHasil, pilihBuat hasil baru. Hasilnya adalah hasil dari prediksi penipuan dan dikembalikan jika aturan cocok selama evaluasi.

4. DalamBuat hasil baru, masuklahverify_customersebagai nama hasil. Secara opsional, masukkan deskripsi.
5. PilihSimpan hasil.
6. PilihTambahkan aturanuntuk menjalankan pemeriksa validasi aturan dan menyimpan aturan. Setelah dibuat, Amazon Fraud Detector membuat aturan tersedia untuk digunakan di detektor Anda.
7. PilihTambahkan aturan lain, dan kemudian pilihBuat aturantab.
8. Ulangi proses ini dua kali lebih banyak untuk membuatmedium_fraud_riskdanlow_fraud_riskaturan menggunakan rincian aturan berikut:

   - medium_fraud_risk

     Nama aturan:medium_fraud_risk

     Hasil:review

     Ekspresi:

     ```
     $sample_fraud_detection_model_insightscore <= 900 and
     ```

     ```
     $sample_fraud_detection_model_insightscore > 700
     ```

   - low_fraud_risk

     Nama aturan:low_fraud_risk

     Hasil:approve

     Ekspresi:

     ```
     $sample_fraud_detection_model_insightscore <= 700
     ```

9. Setelah Anda membuat semua aturan untuk kasus penggunaan Anda, pilihBerikutnya.

   Untuk informasi selengkapnya tentang membuat dan menulis aturan, lihatAturandanReferensi bahasa aturan.

Langkah 4: Konfigurasikan eksekusi aturan dan urutan aturan

Mode eksekusi aturan untuk aturan yang disertakan dalam detektor menentukan apakah semua aturan yang Anda tetapkan dievaluasi, atau jika evaluasi aturan berhenti pada aturan pertama yang cocok. Dan urutan aturan menentukan urutan yang Anda inginkan aturan dijalankan.

Mode eksekusi aturan default adalah`FIRST_MATCHED`.

Pertama cocok

Mode eksekusi aturan pertama yang cocok mengembalikan hasil untuk aturan pencocokan pertama berdasarkan urutan aturan yang ditentukan. Jika Anda menentukan `FIRST_MATCHED`, Amazon Fraud Detector mengevaluasi aturan secara berurutan, pertama hingga terakhir, dengan berhenti pada aturan pertama yang cocok. Amazon Fraud Detector kemudian memberikan hasil untuk aturan tunggal tersebut.

Urutan Anda menjalankan aturan dapat memengaruhi hasil prediksi penipuan yang dihasilkan. Setelah Anda membuat aturan, pesan ulang aturan untuk menjalankannya sesuai urutan yang diinginkan dengan mengikuti langkah-langkah berikut:

Jika Anda`high_fraud_risk`aturan belum ada di bagian atas daftar aturan Anda, pilihPesanan, dan kemudian pilih1. Ini bergerak`high_fraud_risk`ke posisi pertama.

Ulangi proses ini sehingga Anda`medium_fraud_risk`aturan adalah di posisi kedua dan Anda`low_fraud_risk`ada di posisi ketiga.

Semua cocok

Semua mode eksekusi aturan yang cocok mengembalikan hasil untuk semua aturan yang cocok, terlepas dari urutan aturan. Jika Anda menentukan`ALL_MATCHED`, Amazon Fraud Detector mengevaluasi semua aturan dan mengembalikan hasil untuk semua aturan yang cocok.

Pilih`FIRST_MATCHED`untuk tutorial ini dan kemudian pilihBerikutnya.

Langkah 5: Tinjau dan buat versi detektor

Versi detektor mendefinisikan model dan aturan spesifik yang digunakan untuk menghasilkan prediksi penipuan.

1. DalamTinjau dan buathalaman, tinjau detail detektor, model, dan aturan yang Anda konfigurasikan. Jika Anda perlu melakukan perubahan apa pun, pilihMengeditdi sebelah bagian yang sesuai.
2. PilihBuat detektor. Setelah dibuat, versi pertama detektor Anda muncul di tabel versi Detector dengan`Draft`status.

   Anda menggunakanDraftversi untuk menguji Detector Anda.

Buat detektor menggunakanAWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan sampel untuk`PutDetector`API. Detektor bertindak sebagai wadah untuk versi detektor Anda. Yang`PutDetector`API menentukan jenis kejadian apa yang akan dievaluasi oleh detektor. Contoh berikut mengasumsikan Anda telah membuat jenis peristiwa`sample_registration`.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_detector (
detectorId = 'sample_detector',
eventTypeName = 'sample_registration'
)
```

Buat versi detektor

Versi detektor mendefinisikan aturan, perintah eksekusi aturan, dan opsional versi model, yang akan digunakan sebagai bagian dari permintaan untuk menghasilkan prediksi penipuan. Anda dapat menambahkan aturan apa pun yang ditentukan dalam detektor ke versi detektor. Anda juga dapat menambahkan model apa pun yang dilatih pada jenis peristiwa yang dievaluasi.

Setiap versi detektor memiliki status`DRAFT`,`ACTIVE`, atau`INACTIVE`. Hanya satu versi detektor yang bisa masuk`ACTIVE`status pada suatu waktu. Selama`GetEventPrediction`permintaan, Amazon Fraud Detector akan menggunakan`ACTIVE`detektor jika tidak`DetectorVersion`ditentukan.

Mode eksekusi aturan

Amazon Fraud Detector mendukung dua mode eksekusi aturan yang berbeda:`FIRST_MATCHED`dan`ALL_MATCHED`.

- Jika mode eksekusi aturan`FIRST_MATCHED`, Amazon Fraud Detector mengevaluasi aturan secara berurutan, pertama hingga terakhir, berhenti pada aturan pertama yang cocok. Amazon Fraud Detector kemudian memberikan hasil untuk aturan tunggal tersebut. Jika aturan mengevaluasi ke false (tidak cocok), aturan berikutnya dalam daftar dievaluasi.
- Jika mode eksekusi aturan`ALL_MATCHED`, maka semua aturan dalam evaluasi dijalankan secara paralel, terlepas dari urutannya. Amazon Fraud Detector menjalankan semua aturan dan mengembalikan hasil yang ditentukan untuk setiap aturan yang cocok.

Buat versi detektor menggunakanAWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan sampel untuk`CreateDetectorVersion`API. Mode eksekusi aturan diatur ke`FIRST_MATCHED`, oleh karena itu Amazon Fraud Detector akan mengevaluasi aturan secara berurutan, pertama hingga terakhir, berhenti pada aturan pertama yang cocok. Amazon Fraud Detector kemudian memberikan hasil untuk aturan tunggal tersebut selama`GetEventPrediction response`.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_detector_version(
detectorId = 'sample_detector',
rules = [{
    'detectorId' : 'sample_detector',
    'ruleId' : 'high_fraud_risk',
    'ruleVersion' : '1'
},
{
    'detectorId' : 'sample_detector',
    'ruleId' : 'medium_fraud_risk',
    'ruleVersion' : '1'
},
{
    'detectorId' : 'sample_detector',
    'ruleId' : 'low_fraud_risk',
    'ruleVersion' : '1'
}
],
modelVersions = [{
    'modelId' : 'sample_fraud_detection_model',
    'modelType': 'ONLINE_FRAUD_INSIGHTS',
```

```
    'modelVersionNumber' : '1.00'
}],
ruleExecutionMode = 'FIRST_MATCHED'
)
```

Untuk memperbarui status versi detektor, gunakanUpdateDetectorVersionStatusAPI. Contoh berikut memperbarui status versi detektor dariDRAFTkepadaACTIVE. SelamaGetEventPredictionpermintaan, jika ID detektor tidak ditentukan, Amazon Fraud Detector akan menggunakanACTIVEversi detektor.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_detector_version_status(
detectorId = 'sample_detector',
detectorVersionId = '1',
status = 'ACTIVE'
)
```

Menghapus detektor, versi detektor, atau versi aturan

Sebelum menghapus detektor di Amazon Fraud Detector, Anda harus terlebih dahulu menghapus semua versi detektor dan versi aturan yang terkait dengan detektor.

Saat Anda menghapus detektor, versi detektor, atau versi aturan, Amazon Fraud Detector menghapus sumber daya tersebut secara permanen dan data tidak lagi disimpan di Amazon Fraud Detector.

Menghapus versi detektor

Anda hanya dapat menghapus versi detektor yang ada dalamDRAFT atauINACTIVE status.

1. Masuklah keAWS Management Console dan buka konsol Amazon Fraud Detector di https://console.aws.amazon.com/frauddetector.
2. Di panel navigasi kiri konsol Amazon Fraud Detector Amazon, pilih Detector.
3. Pilih detektor yang berisi versi detektor yang ingin Anda hapus.
4. Pilih versi detektor yang ingin Anda hapus.
5. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).

6. Masukkan**`delete`**, lalu pilih Hapus detektor.

Untuk menghapus versi

Anda dapat menghapus versi aturan hanya jika tidak digunakan oleh versi apapun`ACTIVE` atau`INACTIVE` detektor. Jika perlu, sebelum menghapus versi aturan, pertama-tama pindahkan versi`ACTIVE` detektor ke`INACTIVE`, lalu hapus versi`INACTIVE` detektor.

1. Di panel navigasi kiri konsol Amazon Fraud Detector Amazon, pilih Detector.
2. Pilih detektor yang berisi versi yang ingin Anda hapus.
3. Pilih tab Aturan, dan pilih aturan yang ingin Anda hapus.
4. Pilih versi yang ingin Anda hapus.
5. Pilih Tindakan, lalu pilih Hapus versi aturan.
6. Masukkan**`delete`**, lalu pilih Hapus versi.

Menghapus detektor

Sebelum menghapus detektor, Anda harus terlebih dahulu menghapus semua versi detektor dan versi aturan yang terkait dengan detektor.

1. Di panel navigasi kiri konsol Amazon Fraud Detector Amazon, pilih Detector.
2. Pilih detektor yang ingin Anda hapus.
3. Pilih Tindakan, lalu pilih Hapus detektor.
4. Masukkan**`delete`**, lalu pilih Hapus detektor.

Sumber daya

Model, aturan, dan detektor menggunakan sumber daya seperti variabel, hasil, label, daftar, dan entitas untuk mengevaluasi kejadian risiko penipuan. Bagian ini memberikan informasi tentang membuat dan mengelola sumber daya.

Topik

Variabel

Variabel mewakili elemen data yang ingin Anda gunakan dalam prediksi penipuan. Variabel-variabel ini dapat diambil dari kumpulan data peristiwa yang Anda siapkan untuk melatih model Anda, dari output skor risiko model Amazon Fraud Detector, atau dari model Amazon. SageMaker Untuk informasi lebih lanjut tentang variabel yang diambil dari dataset peristiwa, lihat. Dapatkan persyaratan dataset acara menggunakan data model explorer

Variabel yang ingin Anda gunakan dalam prediksi penipuan Anda harus terlebih dahulu dibuat dan kemudian ditambahkan ke acara saat membuat jenis acara Anda. Setiap variabel yang Anda buat harus diberi tipe data, nilai default, dan opsional tipe variabel. Amazon Fraud Detector memperkaya beberapa variabel yang Anda berikan seperti alamat IP, nomor identifikasi bank (BIN), dan nomor telepon, untuk membuat input tambahan dan meningkatkan kinerja untuk model yang menggunakan variabel ini.

Jenis Data

Variabel harus memiliki tipe data untuk elemen data yang variabel mewakili dan opsional dapat ditugaskan salah satu yang telah ditetapkan. Jenis variabel Untuk variabel yang ditugaskan untuk jenis variabel, tipe data pra-dipilih. Jenis data yang mungkin termasuk jenis berikut:

Tipe data	Deskripsi	Nilai default	Contoh nilai
String	Kombinasi huruf, bilangan bulat, atau keduanya	<empty>	abc, 123
Bulat	Bilangan bulat positif atau negatif	0	1, -1
Boolean	Benar atau Salah	Salah	Betul, Salah
DateTime	Tanggal dan waktu yang ditentukan dalam format UTC standar ISO 8601 saja	<empty>	2019-11-30T 13:01:01 Z
Desimal	Angka dengan titik desimal	0.0	4.01, 0,10

Nilai default

Variabel harus memiliki nilai default. Saat Amazon Fraud Detector menghasilkan prediksi penipuan, nilai default ini digunakan untuk menjalankan aturan atau model jika Amazon Fraud Detector tidak menerima nilai untuk variabel. Nilai default yang Anda berikan harus sesuai dengan tipe data yang dipilih. Di Konsol AWS, Amazon Fraud Detector menetapkan nilai default `0` untuk bilangan bulat, untuk Boolean, `false` untuk float, dan (kosong) `0.0` untuk string. Anda dapat menetapkan nilai default kustom untuk salah satu jenis data ini.

Jenis variabel

Bila Anda membuat variabel, Anda opsional dapat menetapkan variabel untuk jenis variabel. Jenis variabel mewakili elemen data umum yang digunakan untuk melatih model dan untuk menghasilkan prediksi penipuan. Hanya variabel dengan tipe variabel terkait yang dapat digunakan untuk pelatihan model. Sebagai bagian dari proses pelatihan model, Amazon Fraud Detector menggunakan jenis variabel yang terkait dengan variabel untuk melakukan pengayaan variabel, rekayasa fitur, dan penilaian risiko.

Amazon Fraud Detector telah menentukan jenis variabel berikut yang dapat digunakan untuk menetapkan ke variabel Anda.

Ka	Tipe variabel	Deskripsi	Tipe data	Co
Ses	IP_ADDRESS	Alamat IP yang dikumpulkan selama acara berlangsung	Strin	192.0.2.0 Catatan: Amazon Fraud Detector memperkaya data ini. Untuk informasi selengkapnya, lihat Pengayaan geolokasi
	AGEN PENGGUNA	Agen pengguna yang dikumpulkan selama acara berlangsung	Strin	Mozilla 5.0 (Windows NT 10.0, Win64, x64, rv:68.0) Tokek 20100101
	SIDIK JARI	Pengenal unik untuk perangkat yang digunakan untuk acara tersebut	Strin	sadfow987u234

Ka	Tipe variabel	Deskripsi	Tipe data	Co
	SESSION_ID	ID sesi untuk sesi aktif acara	Strin	sid123456789
	AD_CREDENTIALS_VALID	Menunjukkan apakah mandat yang digunakan untuk event login valid	Bool	Benar
Pengguna	EMAIL_ALAMAT	Alamat email yang dikumpulkan selama acara berlangsung	Strin	abc@domain.com
	NOMOR TELEPON	Nomor telepon yang dikumpulkan selama acara berlangsung	Strin	+1 555-0100 Catatan: Amazon Fraud Detector memperkaya data ini. Untuk informasi selengkapnya, lihat Pengayaan nomor telepon

Ka	Tipe variabel	Deskripsi	Tipe data	Co
Penagihan	BILLING_NAME	Nama yang terkait dengan alamat penagihan	Strin	John Doe
	BILLING_TELEPON	Nomor telepon yang terkait dengan alamat penagihan	Strin	+1 555-0100 Catatan: Amazon Fraud Detector memperkaya data ini. Untuk informasi selengkapnya, lihat Pengayaan nomor telepon
	BILLING_ADDRESS_L	Baris pertama dari alamat penagihan	Strin	Setiap jalan
	BILLING_ADDRESS_L	Baris kedua dari alamat penagihan	Strin	Unit 123
	BILLING_CITY	Kota yang ada di alamat penagihan	Strin	Kota mana saja

Ka	Tipe variabel	Deskripsi	Tipe data	Co
	BILLING_STATE	Negara bagian atau provinsi yang ada di alamat penagihan	Strin	Setiap negara bagian atau provinsi
	BILLING_COUNTRY	Negara yang ada di alamat penagihan	Strin	Setiap negara Catatan: Amazon Fraud Detector memperkaya data ini. Untuk informasi selengkapnya, lihat Pengayaan geolokasi

Ka	Tipe variabel	Deskripsi	Tipe data	Co
	PENAGIHA _ZIP	Kode pos yang ada di alamat penagihan	Strin	01234 Catatan: Amazon Fraud Detector memperkaya data ini. Untuk informasi selengkapnya, lihat Pengayaan geolokasi
Pe n	SHIPPING_ NAME	Nama yang terkait dengan alamat pengiriman	Strin	John Doe

Ka	Tipe variabel	Deskripsi	Tipe data	Co
	SHIPPING_PHONE	Nomor telepon yang terkait dengan alamat pengiriman	Strin	+1 555-0100 Catatan: Amazon Fraud Detector memperkaya data ini. Untuk informasi selengkapnya, lihat Pengayaan nomor telepon
	SHIPPING_ADDRESS_1	Baris pertama dari alamat pengiriman	Strin	123 Setiap Jalan
	SHIPPING_ADDRESS_2	Baris kedua dari alamat pengiriman	Strin	Satuan 123
	SHIPPING_CITY	Kota yang ada di alamat pengiriman	Strin	Kota mana saja

Ka	Tipe variabel	Deskripsi	Tipe data	Co
	SHIPPING_STATE	Negara bagian atau provinsi yang ada di alamat pengiriman	Strin	Setiap Negara
	SHIPPING_COUNTRY	Negara yang ada di alamat pengiriman	Strin	Setiap Negara Catatan: Amazon Fraud Detector memperkaya data ini. Untuk informasi selengkapnya, lihat Pengayaan geolokasi

Ka	Tipe variabel	Deskripsi	Tipe data	Co
	PENGIRIMAN_ZIP	Kode pos yang ada di alamat pengiriman	Strin	01234 Catatan: Amazon Fraud Detector memperkaya data ini. Untuk informasi selengkapnya, lihat Pengayaan geolokasi
Pembayaran	ORDER_ID	Pengenal unik untuk transaksi	Strin	LUX60
	HARGA	Total harga pesanan	Strin	560.00
	CURRENCY_KODE	Kode mata uang ISO 4217	Strin	USD
	PEMBAYARAN_TYPE	Metode pembayaran yang digunakan untuk pembayaran selama acara berlangsung	Strin	Kartu kredit

Ka	Tipe variabel	Deskripsi	Tipe data	Co
	KODE	Kode alfanumerik yang dikirim oleh penerbit kartu kredit atau bank penerbit	Strin	0000
	AVS	Kode respons sistem verifikasi alamat (AVS) dari prosesor kartu	Strin	Y
Pr	PRODUCT_ ATEGORY	Kategori produk item pesanan	Strin	Dapur
Khusus	NUMERIK	Setiap variabel yang dapat direpresentasikan sebagai bilangan real	Desi	1.224
	KATEGORI	Variabel apa pun yang menjelaskan kategori, segmen, atau grup	Strin	Besar
	FREE_FOR _TEXT	Teks formulir gratis apa pun yang diambil sebagai bagian dari acara (misalnya, ulasan atau komentar pelanggan)	Strin	Contoh input teks formulir bebas

Menetapkan variabel untuk jenis variabel

Jika Anda berencana menggunakan variabel untuk melatih model Anda, penting bagi Anda untuk memilih jenis variabel yang tepat untuk ditetapkan ke variabel. Penetapan jenis variabel yang salah dapat berdampak negatif pada kinerja model Anda. Hal ini juga bisa menjadi sangat sulit bagi Anda mengubah tugas nanti, terutama jika beberapa model dan peristiwa telah menggunakan variabel.

Anda dapat menetapkan variabel Anda salah satu dari jenis variabel yang telah ditentukan sebelumnya atau salah satu jenis variabel kustom -`FREE_FORM_TEXT`,`CATEGORICAL`, atau`NUMERIC`.

Catatan penting untuk menetapkan variabel ke jenis variabel yang tepat

1. Jika variabel cocok dengan salah satu jenis variabel yang telah ditetapkan, menggunakannya. Pastikan jenis variabel sesuai dengan variabel. Misalnya, jika Anda menetapkan variabel ip_address ke tipe variabel, `EMAIL_ADDRESS` variabel ip_address tidak akan diperkaya dengan pengayaan seperti ASN, ISP, geo-location, dan skor risiko. Untuk informasi selengkapnya, lihat Pengayaan variabel.
2. Jika variabel tidak cocok dengan salah satu jenis variabel yang telah ditetapkan, ikuti rekomendasi yang tercantum di bawah ini untuk menetapkan salah satu jenis variabel kustom.
3. Menetapkan jenis `CATEGORICAL` variabel untuk variabel yang biasanya tidak memiliki urutan alami dan dapat dimasukkan ke dalam kategori, segmen, atau kelompok. Dataset yang Anda gunakan untuk melatih model Anda mungkin memiliki variabel ID seperti, merchant_id, campaign_id, atau policy_id. Variabel-variabel ini mewakili kelompok (misalnya, semua pelanggan dengan policy_id yang sama mewakili grup). Variabel yang memiliki data berikut harus ditetapkan CATEGORICAL jenis variabel -
   - Variabel yang berisi data seperti Customer_ID, Segment_ID, Color_ID, department_code, atau Product_ID.
   - Variabel yang berisi data Boolean dengan nilai true, false, atau null.
   - Variabel yang dapat dimasukkan ke dalam kelompok atau kategori seperti nama perusahaan, kategori produk, jenis kartu, atau media rujukan.

> **Note**
>
> `ENTITY_ID`adalah tipe variabel cadangan yang digunakan oleh Amazon Fraud Detector untuk menetapkan ke variabel ENTITY_ID. Variabel ENTITY_ID adalah ID entitas yang memulai tindakan yang ingin Anda evaluasi. Jika Anda membuat tipe model Transaction Fraud Insight (TFI), Anda harus menyediakan variabel ENTITY_ID. Anda

harus memutuskan variabel mana dalam data Anda secara unik mengidentifikasi entitas yang memulai tindakan dan meneruskannya sebagai variabel ENTITY_ID. Tetapkan tipe variabel KATEGORIS ke semua ID lain dalam kumpulan data Anda, jika ada dan jika Anda menggunakannya untuk pelatihan model. Contoh ID lain yang bukan entitas dalam kumpulan data Anda dapat berupa Merchant_ID, Policy_ID, dan Campaign_ID.

4. Menetapkan jenis `FREE_FORM_TEXT` variabel untuk variabel yang berisi blok teks. Contoh jenis variabel FREE_FORM_TEXT adalah - ulasan pengguna, komentar, tanggal, dan kode rujukan. Data FREE_FORM_TEXT berisi beberapa token yang dipisahkan oleh pembatas. Pembatas dapat berupa karakter selain simbol alfa-numerik dan garis bawah. Misalnya, ulasan dan komentar pengguna dapat dipisahkan oleh pembatas “spasi”, tanggal dan kode rujukan dapat menggunakan tanda hubung sebagai pembatas untuk memisahkan awalan, akhiran, dan bagian perantara. Amazon Fraud Detector menggunakan pembatas untuk mengekstrak data dari variabel FREE_FORM_TEXT.
5. Menetapkan NUMERIC jenis variabel untuk variabel yang bilangan real dan memiliki urutan yang melekat. Contoh variabel NUMERIC termasuk day_of_the_week, incident_severity, customer_rating. Meskipun, Anda dapat menetapkan jenis variabel KATEGORIS untuk variabel-variabel ini, kami sangat menyarankan untuk menetapkan semua variabel bilangan real dengan urutan yang melekat pada tipe variabel NUMERIC.

Pengayaan variabel

Amazon Fraud Detector memperkaya beberapa elemen data mentah yang Anda berikan seperti alamat IP, nomor identifikasi bank (BIN), dan nomor telepon, untuk membuat input tambahan dan meningkatkan kinerja untuk model yang menggunakan elemen data ini. Pengayaan membantu mengidentifikasi situasi yang berpotensi mencurigakan dan membantu model untuk menangkap lebih banyak penipuan.

Pengayaan nomor telepon

Amazon Fraud Detector memperkaya data nomor telepon dengan informasi tambahan yang berkaitan dengan geolokasi, operator asli, dan validitas nomor telepon. Pengayaan nomor telepon diaktifkan secara otomatis untuk semua model yang dilatih pada atau setelah 13 Desember 2021 dan memiliki nomor telepon yang menyertakan kode negara (+xxx). Jika Anda telah memasukkan variabel nomor telepon dalam model Anda dan telah melatihnya sebelum 13 Desember 2021, latih ulang model Anda sehingga dapat memanfaatkan pengayaan ini.

Kami sangat menyarankan Anda menggunakan format berikut untuk variabel nomor telepon untuk memastikan bahwa data Anda berhasil diperkaya.

Variabel	Format	Deskripsi
NOMOR TELEPON	Standar E.164	Pastikan untuk menyertakan kode negara (+xxx) dengan nomor telepon.
BILLING_PHONE dan SHIPPING_PHONE	Standar E.164	Pastikan untuk menyertakan kode negara (+xxx) dengan nomor telepon.

Pengayaan geolokasi

Mulai 8 Februari 2022 Amazon Fraud Detector menghitung jarak fisik antara nilai IP_ADDRESS, BILLING_ZIP, dan SHIPPING_ZIP yang Anda berikan untuk suatu peristiwa. Jarak yang dihitung digunakan sebagai input untuk model deteksi penipuan Anda.

Untuk mengaktifkan pengayaan geolokasi, data peristiwa Anda harus menyertakan setidaknya dua dari tiga variabel: IP_ADDRESS, BILLING_ZIP, atau SHIPPING_ZIP. Selain itu, setiap nilai BILLING_ZIP dan SHIPPING_ZIP harus memiliki kode BILLING_COUNTRY yang valid dan kode SHIPPING_COUNTRY masing-masing. Jika Anda memiliki model yang dilatih sebelum 8 Februari 2022 dan itu termasuk variabel-variabel ini, Anda harus melatih ulang model untuk mengaktifkan pengayaan geolokasi.

Jika Amazon Fraud Detector tidak dapat menentukan lokasi yang terkait dengan nilai IP_ADDRESS, BILLING_ZIP, atau SHIPPING_ZIP untuk suatu peristiwa karena data tidak valid, nilai placeholder khusus akan digunakan sebagai gantinya. Misalnya, suatu peristiwa memiliki nilai IP_ADDRESS dan BILLING_ZIP yang valid, tetapi nilai SHIPPING_ZIP tidak valid. Dalam hal ini, pengayaan dilakukan hanya untuk IP_ADDRESS-> BILLING_ZIP. Pengayaan tidak dilakukan untuk IP_ADDRESS-> SHIPPING_ZIP dan BILLING_ZIP-> SHIPPING_ZIP. Sebaliknya, nilai placeholder digunakan di tempat mereka. Tidak masalah apakah pengayaan geolokasi diaktifkan untuk model Anda atau tidak, kinerja model Anda tidak berubah.

Anda dapat memilih keluar dari pengayaan geolokasi dengan memetakan variabel BILLING_ZIP dan SHIPPING_ZIP Anda ke tipe variabel CUSTOM_CATEGORICAL. Mengubah jenis variabel tidak memengaruhi kinerja model Anda.

Format variabel geolokasi

Kami sangat menyarankan agar Anda menggunakan format berikut untuk variabel geolokasi untuk memastikan bahwa data lokasi Anda berhasil diperkaya.

Variabel	Format	Deskripsi
IP_ADDRESS	Alamat IPv4	Misalnya - 1.1.1.1
BILLING_ZIP dan SHIPPING_ZIP	Kode pos ISO 3166-1 alpha-2 untuk negara tertentu	Untuk informasi selengkapnya, lihat bagian Kode negara dan wilayah dalam topik ini.
BILLING_COUNTRY dan SHIPPING_COUNTRY	Kode negara standar ISO 3166-1 alfa-2 dua huruf	Untuk informasi selengkapnya, lihat bagian Kode negara dan wilayah dalam topik ini. Amazon Fraud Detector mencoba mencocokkan semua variasi umum nama suatu negara dengan kode negara standar dua huruf ISO 3166-1. Namun, kami tidak dapat menjamin mereka akan dicocokkan dengan benar.

Kode negara dan wilayah

Tabel berikut menyediakan daftar lengkap negara dan wilayah yang didukung oleh Amazon Fraud Detector untuk pengayaan geolokasi. Setiap negara dan wilayah memiliki kode negara yang ditetapkan (khususnya, kode negara dua huruf ISO 3166-1 alfa-2) dan kode pos.

Format kode pos

- 9 - nomor
- a - surat
- [X] - X adalah opsional. Misalnya, Guersney “GY9 [9] 9aa” berarti “GY9 9aa” dan “GY99 9aa” valid. Gunakan satu format.
- [X/XX] - X atau XX dapat digunakan. Misalnya, Bermuda “aa [aa/99]” berarti “aa aa” dan “aa 99" valid. Gunakan salah satu dari format ini, tetapi jangan gunakan keduanya.
- Beberapa negara memiliki awalan tetap. Misalnya, kode pos untuk Andorra adalah AD999. Ini berarti kode negara harus dimulai dengan huruf AD diikuti oleh tiga angka.

Code	Nama	Kode pos
AD	Andorra	AD999
AR	Antillen Belanda	9999
DI	Austria	9999
AU	Australia	9999
AZ	Azerbaijan	9999
BD	Bangladesh	9999
ADA	Belgium	9999
BG	Bulgaria	9999
BM	Bermuda	aa [aa/99]
OLEH	Belarus	999999

Code	Nama	Kode pos
CA	Canada	a9a
CH	Swiss	9999
CL	Chili	9999999
CO	Kolombia	999999
CR	Kosta Rika	99999
CY	Cyprus	9999
CZ	Ceko	999 99
DE	Germany	99999
DK	Denmark	9999
BERBUAT	Republik Dominika	99999
DZ	Aljazair	99999
EE	Estonia	99999
ES	Spain	99999
FI	Finland	99999
FM	Negara Federasi Mikronesia	99999
FO	Kepulauan Faroe	999
FR	France	99999
GB	Britania Raya	sebuah [Sebuah] 9 [Sebuah/9] 9aa
GG	Guernsey	GY9 [9] 9aa
GL	Greenland	9999

Code	Nama	Kode pos
GP	Guadeloupe	99999
GT	Guatemala	99999
GU	Guam	99999
HR	Croatia	99999
HU	Hungary	9999
YAKNI	Ireland	a99 [A/9] [A/9] [A/9] [A/9]
IM	Pulau Manusia	IM9 [9] 9aa
DI DALAM	India	999999
ADALAH	Islandia	999
IA	Italy	99999
JE	Jersey	JE9 [9] 9aa
JP	Jepang	999-9999
KR	Republik Korea	99999
LI	Liechtenstein	9999
LK	Sri Lanka	99999
LT	Lithuania	99999
LU	Luxembourg	L-9999
LV	Latvia	LV-9999
MC	Monako	99999
MD	Republik Moldova	9999

Code	Nama	Kode pos
MH	Kepulauan Marshall	99999
MK	Makedonia Utara	9999
MP	Kepulauan Mariana Utara	99999
MQ	Matinique	99999
MT	Malta	aaa 9999
MX	Meksiko	99999
SAYA	Malaysia	99999
NL	Netherlands	9999 aa
TIDAK	Norwegia	9999
NZ	Selandia Baru	9999
PH	Filipina	9999
PK	Pakistan	99999
PL	Poland	99-999
PR	Puerto Riko	99999
PT	Portugal	9999-999
PW	Palau	99999
KEMBALI	Reuni	99999
RO	Romania	999999
RU	Federasi Rusia	999999
SE	Sweden	999 99

Code	Nama	Kode pos
SG	Singapura	999999
SI	Slovenia	9999
SK	Slovakia	999 99
SM	San Marino	99999
TH	Thailand	99999
TR	Turki	99999
UA	Ukraina	99999
US	Amerika Serikat	99999
UY	Uruguay	99999
VI	Kepulauan Virgin, AS	99999
WF	Wallis dan Futuna	99999
YT	Mayotte	99999
ZA	Afrika Selatan	9999

Pengayaan Useragent

Jika Anda membuat model Account Takeover Insights (ATI), Anda harus menyediakan variabel jenis `useragent` variabel dalam set data Anda. Variabel ini berisi data browser, perangkat, dan OS dari peristiwa login. Amazon Fraud Detector memperkaya data agen pengguna dengan informasi tambahan seperti, dan. `user_agent_family` `OS_family` `device_family`

Buat variabel

Anda dapat membuat variabel di konsol Amazon Fraud Detector, menggunakan perintah create-variable, menggunakan, atau menggunakan CreateVariableAWS SDK for Python (Boto3)

Membuat variabel menggunakan konsol Amazon Fraud Detector

Contoh ini menciptakan dua variabel, `email_address` dan`ip_address`, dan memberikan mereka ke jenis variabel yang sesuai (`EMAIL_ADDRESS`dan`IP_ADDRESS`). Variabel-variabel ini digunakan sebagai contoh. Jika Anda membuat variabel yang akan digunakan untuk pelatihan model Anda, gunakan variabel dari kumpulan data yang sesuai untuk kasus penggunaan Anda. Pastikan untuk membaca tentang Jenis variabel dan Pengayaan variabel sebelum Anda membuat variabel Anda.

Untuk membuat variabel,

1. Buka AWSManagement Console dan masuk ke akun Anda.
2. Arahkan ke Amazon Fraud Detector, pilih Variabel di navigasi kiri, lalu pilih Buat.
3. Di halaman variabel baru, masukkan `email_address` sebagai nama variabel. Opsional, masukkan deskripsi variabel.
4. Dalam jenis Variabel, pilih Alamat Email.
5. Amazon Fraud Detector secara otomatis memilih tipe data untuk jenis variabel ini karena jenis variabel ini telah ditentukan sebelumnya. Jika variabel Anda tidak secara otomatis ditetapkan jenis variabel, pilih jenis variabel dari daftar. Untuk informasi selengkapnya, lihat Jenis variabel.
6. Jika Anda ingin memberikan nilai default untuk variabel Anda, pilih Tentukan nilai default kustom dan masukkan nilai default untuk variabel Anda. Lewati langkah ini jika Anda mengikuti contoh ini.
7. Pilih Create (Buat).
8. Di halaman ikhtisar email_address, konfirmasikan detail variabel yang baru saja Anda buat.

   Jika Anda perlu memperbarui, pilih Edit dan berikan pembaruan. Pilih Save changes (Simpan perubahan).
9. Ulangi proses untuk membuat variabel lain `ip_address` dan pilih Alamat IP untuk jenis variabel.
10. Halaman Variabel menunjukkan variabel yang baru dibuat.

> **Important**
>
> Kami menyarankan Anda membuat variabel sebanyak yang Anda inginkan dari dataset Anda. Anda dapat memutuskan nanti saat membuat jenis acara variabel mana yang ingin Anda sertakan untuk melatih model Anda untuk mendeteksi penipuan dan menghasilkan deteksi penipuan.

Buat variabel menggunakan AWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan untuk CreateVariableAPI. Contoh menciptakan dua variabel, `email_address` dan`ip_address`, dan memberikan mereka ke jenis variabel yang sesuai (`EMAIL_ADDRESS`dan`IP_ADDRESS`).

Variabel-variabel ini digunakan sebagai contoh. Jika Anda membuat variabel yang akan digunakan untuk pelatihan model Anda, gunakan variabel dari kumpulan data yang sesuai untuk kasus penggunaan Anda. Pastikan untuk membaca tentang Jenis variabel dan Pengayaan variabel sebelum Anda membuat variabel Anda.

Pastikan untuk menentukan sumber variabel. Ini membantu untuk mengidentifikasi di mana nilai variabel diturunkan. Jika sumber variabel adalah EVENT, nilai variabel dikirim sebagai bagian dari GetEventPredictionpermintaan. Jika nilai variabelnya`MODEL_SCORE`, itu diisi oleh Amazon Fraud Detector. Jika`EXTERNAL_MODEL_SCORE`, nilai variabel diisi oleh SageMaker model yang diimpor.

```
import boto3
fraudDetector = boto3.client('frauddetector')

 #Create variable email_address
   fraudDetector.create_variable(
     name = 'email_address',
     variableType = 'EMAIL_ADDRESS',
     dataSource = 'EVENT',
     dataType = 'STRING',
     defaultValue = '<unknown>'
     )

#Create variable ip_address
   fraudDetector.create_variable(
     name = 'ip_address',
     variableType = 'IP_ADDRESS',
     dataSource = 'EVENT',
     dataType = 'STRING',
     defaultValue = '<unknown>'
     )
```

Menghapus variabel

Saat Anda menghapus variabel, Amazon Fraud Detector menghapus variabel tersebut secara permanen dan data tidak lagi disimpan di Amazon Fraud Detector.

Anda tidak dapat menghapus variabel yang disertakan dalam jenis peristiwa di Amazon Fraud Detector. Anda harus terlebih dahulu menghapus jenis acara variabel dikaitkan dengan dan kemudian menghapus variabel.

Anda tidak dapat menghapus variabel keluaran model Amazon Fraud Detector dan variabel keluaran SageMaker model secara manual. Amazon Fraud Detector secara otomatis menghapus variabel keluaran model saat Anda menghapus model.

Anda dapat menghapus variabel di konsol Amazon Fraud Detector, menggunakan perintah CLI delete-variable, menggunakan DeleteVariableAPI, atau menggunakan AWS SDK for Python (Boto3)

Hapus variabel menggunakan konsol

Untuk menghapus variabel,

1. Masuk ke AWS Management Console dan buka konsol Amazon Fraud Detector di https://console.aws.amazon.com/frauddetector.
2. Di panel navigasi kiri konsol Amazon Fraud Detector, pilih Sumber Daya, lalu pilih Variabel.
3. Pilih variabel yang ingin Anda hapus.
4. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
5. Masukkan nama variabel, dan kemudian pilih Hapus variabel.

Hapus variabel menggunakan AWS SDK for Python (Boto3)

Contoh kode berikut menghapus variabel customer_name menggunakan API. DeleteVariable

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.delete_variable (

name = 'customer_name'

)
```

Label

Label mengklasifikasikan peristiwa sebagai penipuan atau sah. Label dikaitkan dengan jenis peristiwa dan digunakan untuk melatih model machine learning di Amazon Fraud Detector. Jika Anda berencana untuk melatih model Wawasan Penipuan Online (OFI) atau Transaction Fraud Insights (TFI), minimal 400 peristiwa dalam kumpulan data pelatihan Anda harus diklasifikasikan sebagai penipuan atau sah. Anda dapat menggunakan label apa pun seperti penipuan, legit, 1, atau 0 untuk mengklasifikasikan peristiwa dalam kumpulan data pelatihan Anda. Setelah pelatihan selesai, model terlatih mengevaluasi peristiwa untuk penipuan dan menggunakan nilai-nilai ini untuk mengklasifikasikan peristiwa sebagai penipuan atau sah.

Anda harus terlebih dahulu membuat label dengan nilai yang digunakan dalam kumpulan data pelatihan Anda dan kemudian mengaitkan label dengan jenis peristiwa yang digunakan untuk membuat dan melatih model deteksi penipuan Anda.

Buat label

Anda dapat membuat label di konsol Amazon Fraud Detector, menggunakan perintah put-label, menggunakan PutLabelAPI, atau menggunakanAWS SDK for Python (Boto3).

Membuat label menggunakan konsol Amazon Fraud Detector

Untuk membuat label,

1. AWSLanjutkan ke akun Anda.
2. Arahkan ke Amazon Fraud Detector, pilih Label di navigasi kiri, lalu pilih Buat.
3. Di halaman Buat label, masukkan nama label Anda untuk acara penipuan sebagai nama label. Nama label harus sesuai dengan label yang mewakili aktivitas penipuan dalam kumpulan data pelatihan Anda. Atau, masukkan deskripsi label.
4. Pilih Buat label.
5. Buat label kedua dan masukkan nama label untuk acara yang sah. Pastikan nama label sesuai dengan nilai yang mewakili aktivitas yang sah dalam kumpulan data latihan Anda.

Buat label menggunakanAWS SDK for Python (Boto3)

AWS SDK for Python (Boto3)Contoh kode berikut membuat dua label (penipuan, legit) menggunakan PutLabelAPI. Setelah membuat label, Anda dapat menambahkannya ke jenis acara untuk mengklasifikasikan peristiwa tertentu.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_label(
name = 'fraud',
description = 'label for fraud events'
)

fraudDetector.put_label(
name = 'legit',
description = 'label for legitimate events'
)
```

Perbarui label

Jika kumpulan data peristiwa Anda disimpan dengan Amazon Fraud Detector, Anda mungkin perlu menambahkan atau memperbarui label untuk peristiwa yang disimpan, seperti saat Anda melakukan investigasi penipuan offline untuk suatu peristiwa dan ingin menutup loop balik umpan balik machine learning.

Anda dapat menambahkan atau memperbarui label untuk peristiwa yang disimpan menggunakan update-event-labelperintah, menggunakan UpdateEventLabelAPI, atau menggunakanAWS SDK for Python (Boto3)

AWS SDK for Python (Boto3)Contoh kode berikut menambahkan penipuan label yang terkait dengan pendaftaran jenis peristiwa menggunakan`UpdateEventLabel` API.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_event_label(
            eventId        = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
            eventTypeName  = 'registration',
            assignedLabel  = 'fraud',
            labelTimestamp = '2020-07-13T23:18:21Z'
)
```

Memperbarui label peristiwa dalam data peristiwa yang disimpan di Amazon Fraud Detector

Anda mungkin perlu menambahkan atau memperbarui label penipuan untuk peristiwa yang sudah disimpan di Amazon Fraud Detector, seperti saat Anda melakukan investigasi penipuan offline untuk suatu peristiwa dan ingin menutup loop balik umpan balik machine learning. Untuk memperbarui label peristiwa yang sudah disimpan di Amazon Fraud Detector, gunakan operasi`UpdateEventLabel` API. Berikut ini menunjukkan contoh UpdateEventLabel API panggilan.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_event_label(
            eventId        = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
            eventTypeName  = 'sample_registration',
            assignedLabel  = 'fraud',
            labelTimestamp = '2020-07-13T23:18:21Z'
)
```

Hapus label

Saat Anda menghapus label, Amazon Fraud Detector menghapus label tersebut secara permanen dan data tidak lagi disimpan di Amazon Fraud Detector.

Anda tidak dapat menghapus label yang disertakan dalam jenis peristiwa di Amazon Fraud Detector. Dan Anda juga tidak dapat menghapus label yang ditetapkan ke ID peristiwa. Anda harus terlebih dahulu menghapus ID peristiwa yang relevan.

Anda dapat menghapus label di konsol Amazon Fraud Detector, menggunakan perintah delete-label, menggunakan DeleteLabelAPI, atau menggunakanAWS SDK for Python (Boto3)

Hapus label menggunakan konsol

Untuk menghapus label

1. Masuk keAWS Management Console dan buka konsol Amazon Fraud Detector di https://console.aws.amazon.com/frauddetector.
2. Di panel navigasi kiri konsol Amazon Fraud Detector, pilih Sumber Daya, lalu pilih Label.

3. Pilih label yang ingin Anda hapus.
4. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
5. Masukkan nama label, lalu pilih Hapus label.

Menghapus label menggunakanAWS SDK for Python (Boto3)

AWS SDK for Python (Boto3)Contoh kode berikut menghapus label legit menggunakan DeleteLabelAPI.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.delete_event_label (
    name = 'legit'
)
```

Aturan

Aturan adalah kondisi yang memberi tahu Amazon Fraud Detector cara menafsirkan nilai variabel selama prediksi penipuan. Aturan adalah bagian dari logika detektor dan terdiri dari elemen-elemen berikut:

- Variabel atau Daftar - Variabel mewakili elemen data dalam kumpulan data acara Anda yang ingin Anda gunakan dalam prediksi penipuan. Daftar adalah satu set elemen data masukan untuk variabel dalam dataset acara Anda. Variabel yang digunakan dalam aturan harus ditentukan sebelumnya dalam jenis peristiwa yang dievaluasi dan daftar yang digunakan dalam aturan harus dikaitkan dengan tipe variabel. Untuk informasi selengkapnya, lihat Variabel dan Daftar.
- Ekspresi - Ekspresi dalam aturan menangkap logika bisnis Anda. Jika Anda menggunakan variabel dalam aturan Anda, ekspresi aturan sederhana dibangun menggunakan variabel, operator perbandingan seperti>, <, <=, >=. ==, dan nilai. Jika Anda menggunakan daftar, ekspresi aturan dibangun sebagai entri daftar,`in`, dan nama daftar. Untuk informasi selengkapnya, lihat Referensi bahasa aturan. Anda dapat menggabungkan beberapa ekspresi bersama-sama menggunakan `and` dan`or`. Semua ekspresi harus mengevaluasi ke nilai Boolean (true atau false) dan kurang dari 4.000 karakter panjangnya. Jika-lain kondisi jenis tidak didukung.
- Hasil — Hasil adalah respons yang dikembalikan oleh Amazon Fraud Detector saat aturan dicocokkan. Hasilnya menunjukkan hasil prediksi penipuan. Anda dapat membuat hasil untuk

setiap prediksi penipuan yang mungkin dan menambahkannya ke aturan. Untuk informasi selengkapnya, lihat Hasil.

Detektor harus memiliki setidaknya satu aturan terkait. Aturan dapat memiliki hingga 3 daftar, dan detektor dapat memiliki hingga 30 daftar. Anda membuat aturan sebagai bagian dari proses pembuatan detektor. Anda juga dapat membuat dan mengaitkan aturan baru dengan detektor yang ada.

Referensi bahasa aturan

Bagian berikut menguraikan kemampuan ekspresi (yaitu, penulisan aturan) di Amazon Fraud Detector.

Menggunakan variabel

Anda dapat menggunakan variabel apa pun yang didefinisikan dalam jenis peristiwa yang dievaluasi sebagai bagian dari ekspresi Anda. Gunakan tanda dolar untuk menunjukkan variabel:

```
$example_variable < 100
```

Menggunakan daftar

Anda dapat menggunakan daftar apa pun yang dikaitkan dengan jenis variabel dan diisi dengan entri sebagai bagian dari ekspresi aturan Anda. Gunakan tanda dolar untuk menunjukkan nilai entri daftar:

```
$example_list_variable in @list_name
```

Perbandingan, keanggotaan, dan operator identitas

Amazon Fraud Detector mencakup operator perbandingan berikut: **>**, **>=**, **<**, **<=**,! =, ==, di, tidak di

Berikut ini adalah contoh-contohnya:

Contoh: **<**

```
$variable < 100
```

Contoh: di, tidak di

```
$variable in [5, 10, 25, 100]
```

Contoh:! =

```
$variable != "US"
```

Contoh: ==

```
$variable == 1000
```

Tabel Operator

Operator	Operator Detektor Penipuan Amazon
Sama dengan	==
Tidak sama dengan	!=
Lebih besar dari	>
Kurang dari	<
Besar dari atau sama dengan	>=
Kurang dari atau sama dengan	<=
Di	in
dan	and
Atau	or
Tidak	!

Matematika dasar

Anda dapat menggunakan operator matematika dasar dalam ekspresi Anda (misalnya, +, -, *,/). Kasus penggunaan tipikal adalah ketika Anda perlu menggabungkan variabel selama evaluasi Anda.

Dalam aturan di bawah ini, kita menambahkan variabel `$variable_1` dengan`$variable_2`, dan memeriksa apakah totalnya kurang dari 10.

```
$variable_1 + $variable_2 < 10
```

Data Tabel Matematika Dasar

Operator	Operator Detektor Penipuan Amazon
Ditambah	+
Minus	-
Kalikan	*
Membagi	/
Modulo	%

Ekspresi Reguler (regex)

Anda dapat menggunakan regex untuk mencari pola tertentu sebagai bagian dari ekspresi Anda. Hal ini sangat berguna jika Anda mencari untuk mencocokkan string tertentu atau nilai numerik untuk salah satu variabel Anda. Amazon Fraud Detector hanya mendukung kecocokan saat bekerja dengan ekspresi reguler (misalnya, Amazon Fraud Detector mengembalikan True/False tergantung pada apakah string yang disediakan dicocokkan dengan ekspresi reguler). Dukungan ekspresi reguler Amazon Fraud Detector didasarkan pada .matches () di java (menggunakan pustaka Regular Expression RE2J). Ada beberapa situs web bermanfaat di internet yang berguna untuk menguji pola ekspresi reguler yang berbeda.

Pada contoh pertama di bawah ini, pertama-tama kita mengubah variabel `email` menjadi huruf kecil. Kami kemudian memeriksa `@gmail.com` apakah pola dalam `email` variabel. Perhatikan periode kedua lolos sehingga kita dapat secara eksplisit memeriksa string. `.com`

```
regex_match(".*@gmail\.com", lowercase($email))
```

Pada contoh kedua, kita memeriksa apakah variabel `phone_number` berisi kode negara +1 untuk menentukan apakah nomor telepon dari AS. Simbol plus lolos sehingga kita dapat secara eksplisit memeriksa string. +1

```
regex_match(".*\+1", $phone_number)
```

Tabel Regex

Operator	Contoh Detektor Penipuan Amazon
Cocokkan string yang dimulai dengan	regex_match (“^mystring”, $ variabel)
Cocokkan seluruh string dengan tepat	regex_match (“mystring”, $ variabel)
Cocokkan karakter apa pun kecuali baris baru	regex_match (” . “, $ variabel)
Cocokkan sejumlah karakter kecuali baris baru sebelum 'mystring'	regex_match (”. * mystring “, $ variabel)
Melarikan diri karakter khusus	\

Memeriksa nilai yang hilang

Terkadang bermanfaat untuk memeriksa apakah nilainya hilang. Di Amazon Fraud Detector, ini diwakili oleh null. Anda dapat melakukan ini dengan menggunakan sintaks berikut:

```
$variable != null
```

Demikian pula, jika Anda ingin memeriksa apakah nilai tidak ada, Anda dapat melakukan hal berikut:

```
$variable == null
```

Beberapa kondisi

Anda dapat menggabungkan beberapa ekspresi bersama-sama menggunakan `and` dan`or`. Amazon Fraud Detector berhenti dalam `OR` ekspresi ketika satu nilai sebenarnya ditemukan, dan berhenti `AND` ketika satu nilai palsu ditemukan.

Pada contoh di bawah ini, kami memeriksa dua kondisi menggunakan `and` kondisi. Dalam pernyataan pertama, kita memeriksa apakah variabel 1 kurang dari 100. Pada detik kita memeriksa apakah variabel 2 bukan AS.

Mengingat aturan menggunakan`and`, keduanya harus TRUE untuk seluruh kondisi untuk mengevaluasi ke TRUE.

```
$variable_1 < 100 and $variable_2 != "US"
```

Anda dapat menggunakan tanda kurung untuk mengelompokkan operasi Boolean, seperti yang ditunjukkan berikut:

```
$variable_1 < 100 and $variable_2 != "US" or ($variable_1 * 100.0 > $variable_3)
```

Jenis ekspresi lainnya

DateTime fungsi

Fungsi	Deskripsi	Contoh
getcurrentdatetime ()	Memberikan waktu eksekusi aturan saat ini dalam format UTC ISO8601. Anda dapat menggunakan getepochmilliseconds (getcurrentdatetime ()) untuk melakukan operasi tambahan	getcurrentdatetime () == “2023-03-28T 18:34:02 Z”
sebelum (DateTime1, DateTime 2)	Mengembalikan boolean (True/False) jika pemanggil 1 sebelum 2 DateTime DateTime	isbefore (getcurrentdatetime (), “2019-11-30T 01:01:01 Z”) == “Salah” isbefore (getcurrentdatetime (), “2050-11-30T 01:05:01 Z”) == “Benar”
setelah (DateTime1, DateTime 2)	Mengembalikan boolean (True/False) jika pemanggil 1 adalah setelah 2 DateTime DateTime	isafter (getcurrentdatetime (), “2019-11-30T 01:01:01 Z”) == “Benar” isafter (getcurrentdatetime (), “2050-11-30T 01:05:01 Z”) == “Salah”
getepochmilidetik () DateTime	Membawa DateTime dan mengembalikan bahwa DateTime	getepochmilidetik (“2019-11-30T 01:01:01 Z”) == 1575032461

Fungsi	Deskripsi	Contoh
	dalam milidetik epoch. Berguna untuk melakukan operasi matematik a pada tanggal	

Operator String

Operator	Contoh
Transform string ke huruf besar	huruf besar ($ variabel)
Transform string ke huruf kecil	huruf kecil ($ variabel)

Lainnya

Operator	Komentar
Tambahkan komentar	# komentar saya

Buat aturan

Anda dapat membuat aturan di konsol Amazon Fraud Detector, menggunakan perintah create-rule, menggunakan CreateRuleAPI, atau menggunakan. AWS SDK for Python (Boto3)

Setiap aturan harus berisi ekspresi tunggal yang menangkap logika bisnis Anda. Semua ekspresi harus mengevaluasi ke nilai Boolean (true atau false) dan kurang dari 4.000 karakter panjangnya. Jika-lain kondisi jenis tidak didukung. Semua variabel yang digunakan dalam ekspresi harus ditetapkan sebelumnya dalam jenis acara dievaluasi. Demikian pula, semua daftar yang digunakan dalam ekspresi harus ditentukan sebelumnya, terkait dengan tipe varible, dan diisi dengan entri.

Contoh berikut membuat aturan `high_risk` untuk detektor yang ada`payments_detector`. Aturan tersebut mengaitkan ekspresi dan hasil `verify_customer` dengan aturan.

Prasyarat

Untuk mengikuti langkah-langkah yang disebutkan di bawah ini, pastikan Anda menyelesaikan hal berikut sebelum melanjutkan dengan membuat aturan:

- Buat detektor
- Buat hasil

Jika Anda membuat detektor, aturan, dan hasil untuk kasus penggunaan, ganti nama detektor contoh, nama aturan, ekspresi aturan, dan nama hasil dengan nama dan ekspresi yang relevan dengan kasus penggunaan Anda.

Membuat aturan baru di konsol Amazon Fraud Detector

1. Buka AWSManagement Console dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Detectors dan pilih detektor yang Anda buat untuk kasus penggunaan Anda, misalnya payments_detector.
3. Di halaman payments_detector, pilih tab Aturan terkait, lalu pilih Buat aturan.
4. Di halaman Aturan baru, masukkan yang berikut ini:
    a. Dalam Nama, masukkan nama untuk aturan, contoh **`high_risk`**
    b. Dalam Deskripsi - opsional, secara opsional masukkan deskripsi aturan, misalnya, **`This rule captures events with a high ML model score`**
    c. Dalam Ekspresi, masukkan ekspresi aturan untuk kasus penggunaan Anda menggunakan panduan referensi cepat Ekspresi. Contoh `$sample_fraud_detection_model_insightscore >900`
    d. Dalam Hasil, pilih hasil yang Anda buat untuk kasus penggunaan Anda, misalnya verify_customer. Hasilnya adalah hasil dari prediksi penipuan dan dikembalikan jika aturan cocok selama evaluasi.
5. Pilih Simpan aturan

Anda membuat aturan baru untuk detektor Anda. Ini adalah versi 1 dari aturan yang Amazon Fraud Detector secara otomatis membuatnya tersedia untuk digunakan detektor.

Buat aturan menggunakan AWS SDK for Python (Boto3)

Kode contoh berikut menggunakan CreateRuleAPI untuk membuat aturan `high_risk` untuk detektor yang ada`payments_detector`. Kode contoh juga menambahkan ekspresi aturan dan hasil `verify_customer` ke aturan.

Prasyarat

Untuk menggunakan kode contoh, pastikan Anda telah menyelesaikan hal-hal berikut sebelum melanjutkan dengan membuat aturan:

- Buat detektor
- Buat hasil

Jika Anda membuat detektor, aturan, dan hasil untuk kasus penggunaan, ganti nama detektor contoh, nama aturan, ekspresi aturan, dan nama hasil dengan nama dan ekspresi yang relevan dengan kasus penggunaan Anda.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_rule(
ruleId = 'high_risk',
detectorId = 'payments_detector',
expression = '$sample_fraud_detection_model_insightscore > 900',
language = 'DETECTORPL',
outcomes = ['verify_customer']
)
```

Anda telah membuat versi 1 dari aturan yang secara otomatis membuat Amazon Fraud Detector tersedia untuk digunakan detektor.

Perbarui aturan

Anda dapat memperbarui aturan kapan saja dengan menambahkan atau memperbarui deskripsi aturan, memperbarui ekspresi aturan, atau menambahkan atau menghapus hasil aturan. Ketika Anda memperbarui aturan versi aturan baru dibuat.

Anda dapat memperbarui aturan di konsol Amazon Fraud Detector, menggunakan update-rule-versionperintah, menggunakan UpdateRuleVersionAPI, atau menggunakan AWS SDK.

Setelah Anda memperbarui aturan, pastikan untuk memperbarui versi detektor untuk menggunakan versi aturan baru.

Perbarui aturan di konsol Amazon Fraud Detector

Untuk memperbarui aturan,

1. Buka AWSManagement Console dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Detektor.
3. Di panel Detectors, pilih detektor yang terkait dengan aturan yang ingin Anda perbarui.
4. Di halaman detektor, pilih tab Aturan terkait dan pilih aturan yang ingin Anda perbarui.
5. Di halaman aturan Anda, pilih Tindakan dan pilih Buat versi.
6. Perhatikan bahwa versi telah berubah. Masukkan deskripsi, ekspresi, atau hasil yang diperbarui.
7. Pilih Simpan versi baru

Perbarui aturan menggunakan AWS SDK for Python (Boto3)

Kode contoh berikut menggunakan UpdateRuleVersionAPI untuk memperbarui ambang batas aturan `high_risk` dari 900 ke 950. Aturan ini dikaitkan dengan detektor`payments_detector`.

```
fraudDetector.update_rule_version(
rule = {
    'detectorId' : 'payments_detector',
    'ruleId' : 'high_risk',
    'ruleVersion' : '1'
},
expression = '$sample_fraud_detection_model_insightscore > 950',
language = 'DETECTORPL',
outcomes = ['verify_customer']
)
```

Daftar

Daftar adalah sekumpulan data input untuk variabel dalam kumpulan data acara Anda. Anda menggunakan data input dalam aturan yang terkait dengan detektor Anda. Aturan adalah suatu kondisi yang memberitahu Amazon Fraud Detector mengenai cara menafsirkan data input selama prediksi penipuan. Misalnya, Anda dapat membuat daftar alamat IP dan kemudian membuat aturan

untuk menolak akses jika alamat IP tertentu ada dalam daftar. Aturan yang menggunakan daftar dinyatakan`$ip_address_value` dalam`@list_name` format.

Dengan Amazon Fraud Detector, Anda dapat mengelola daftar dengan menambahkan atau menghapus data tanpa perlu memperbarui aturan terkait. Aturan yang terkait dengan daftar Anda secara otomatis menggabungkan data yang baru ditambahkan atau dihapus.

Daftar dapat berisi hingga 100.000 entri unik dan setiap entri dapat mencapai 320 karakter. Setiap daftar yang Anda gunakan dalam aturan, secara default, terkait denganJenis variabel FREE_FORM_TEXT Amazon Fraud Detector. Anda dapat menetapkan jenis variabel ke daftar Anda kapan saja. Anda dapat menggunakan hingga 3 daftar dalam aturan.

Anda dapat membuat daftar, menambahkan entri ke daftar, menghapus daftar, atau menghapus satu atau beberapa entri dalam daftar, atau menetapkan jenis variabel ke daftar Anda di konsol Amazon Fraud Detector, menggunakan API, menggunakanAWS CLI, atau menggunakanAWS SDK.

Buat daftar

Anda dapat membuat daftar yang berisi data input (entri) variabel dalam kumpulan data acara Anda dan menggunakan daftar dalam ekspresi aturan. Entri dalam daftar dapat dikelola secara dinamis tanpa memperbarui aturan yang menggunakan daftar.

Untuk membuat daftar, Anda harus terlebih dahulu menentukan nama dan kemudian secara opsional mengaitkan daftar denganJenis variabel didukung oleh Amazon Fraud Detector. Secara default, Amazon Fraud Detector mengasumsikan daftar tersebut adalah tipe variabel FREE_FORM_TEXT.

Anda dapat membuat daftar di konsol Amazon Fraud Detector, menggunakan API, menggunakanAWS CLI, atau menggunakanAWS SDK.

Buat daftar menggunakan konsol Amazon Fraud Detector

Untuk membuat daftar

1. Luncurkan AWSManagement Console dan masuk ke akun Anda. Luncurkan ke Amazon Fraud Detector.
2. Di panel navigasi sebelah kiri, pilih Daftar.
3. Di bawah Daftar rincian
    a. Di Nama daftar, masukkan nama untuk daftar Anda.
    b. Di Deskripsi, secara opsional, masukkan deskriptif.

c. (Opsional) Dalam jenis Variabel, pilih jenis variabel untuk daftar Anda.

> **Important**
>
> Jika daftar Anda berisi alamat IP, pastikan untuk memilih IP_ADDRESS sebagai jenis variabel. Jika Anda tidak memilih jenis variabel, Amazon Fraud Detector mengasumsikan daftar tersebut adalah tipe variabel FREE_FORM_TEXT.

4. Dalam Tambahkan data daftar, tambahkan entri daftar, satu entri di setiap baris. Anda juga dapat menyalin dan menempelkan entri dari spreadsheet.

> **Note**
>
> Pastikan bahwa entri tidak dipisahkan menggunakan koma dan unik dalam daftar. Jika dua entri identik dimasukkan, hanya satu yang akan ditambahkan.

5. Pilih Create (Buat).

Buat daftar menggunakanAWS SDK for Python (Boto3)

Anda membuat daftar dengan menentukan nama daftar. Anda dapat secara opsional memberikan deskripsi, mengaitkan jenis variabel, atau menambahkan entri ke daftar Anda saat Anda membuat daftar. Atau, Anda dapat memperbarui daftar nanti dengan menambahkan entri atau deskripsi. Anda dapat menetapkan jenis variabel ke daftar nanti jika Anda belum menetapkannya ketika pada saat pembuatan daftar. Jenis variabel daftar tidak dapat diubah setelah ditugaskan.

> **Important**
>
> Jika daftar Anda berisi alamat IP, pastikan untuk menetapkan IP_ADDRESS sebagai jenis variabel. Jika Anda tidak menetapkan jenis variabel, Amazon Fraud Detector mengasumsikan daftar tersebut adalah tipe variabel FREE_FORM_TEXT.

Contoh berikut menggunakan operasi CreateListAPI untuk membuat`allow_email_ids` daftar dengan memberikan deskripsi, jenis variabel, dan dengan menambahkan empat entri daftar.

```
import boto3
fraudDetector = boto3.client('frauddetector')
```

```
fraudDetector.create_list (
     name = 'allow_email_ids',
     description = 'legitimate email_ids'
     variableType = 'EMAIL_ADDRESS',
     elements = ['emailId _1', 'emailId_2', 'emailId_3','emailId_4']
     )
```

Menambahkan entri dalam daftar

Setelah Anda membuat daftar Anda, Anda dapat menambahkan atau menambahkan entri dalam daftar Anda kapan saja. Saat menambahkan atau menambahkan entri dalam daftar, Anda tidak perlu memperbarui aturan yang terkait dengan daftar. Aturan secara otomatis menggabungkan entri yang baru ditambahkan.

Daftar Anda dapat berisi hingga 100.000 entri unik dan setiap entri dapat mencapai 320 karakter.

Anda dapat menambahkan entri di konsol Amazon Fraud Detector, menggunakan API, menggunakanAWS CLI, atau menggunakanAWS SDK.

Menambahkan entri dalam daftar menggunakan konsol Amazon Fraud Detector

Untuk menambahkan satu atau beberapa entri dalam daftar

1. Luncurkan AWSManagement Console dan masuk ke akun Anda. Luncurkan ke Amazon Fraud Detector.
2. Di panel navigasi sebelah kiri, pilih Daftar.
3. Di halaman Daftar, pilih daftar yang ingin Anda tambahkan entri.
4. Di halaman detail daftar Anda, pilih tab Daftar data dan pilih Tambahkan data.
5. Dalam Tambahkan data daftar kotak, tambahkan satu entri pada setiap baris atau salin dan tempel entri dari spreadsheet. Pastikan untuk tidak menggunakan koma untuk memisahkan entri.
6. Pilih Tambahkan.

Tambahkan entri dalam daftar menggunakanAWS SDK for Python (Boto3)

Contoh berikut menggunakan operasi UpdateListAPI untuk menambahkan dua entri baru dalam`allow_email_ids` daftar. Pastikan bahwa entri yang Anda tambahkan unik dalam daftar.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_list (
     name = 'allow_email_ids',
     updateMode = 'APPEND'
     elements = ['emailId_11','emailId_12']
```

Menetapkan jenis variabel ke daftar

Setiap daftar yang Anda gunakan dalam aturan harus dikaitkan dengan jenisJenis variabel variabel Amazon Fraud Detector. Secara default, Amazon Fraud Detector mengasumsikan daftar tersebut adalah tipe variabel FREE_FORM_TEXT. Penting untuk dicatat bahwa daftar yang terdiri dari alamat IP harus dikaitkan dengan tipe variabel IP_ADDRESS.

Anda dapat mengaitkan daftar Anda dengan jenis variabel baik pada saat pembuatan daftar atau kapan saja nanti. Jika Anda sudah mengaitkan daftar Anda dengan tipe variabel dan ingin mengubahnya nanti, Anda harus membuat daftar baru. Anda tidak dapat mengubah jenis variabel daftar.

Anda dapat menetapkan jenis variabel di konsol Amazon Fraud Detector, menggunakan API, menggunakanAWS CLI, atau menggunakanAWS SDK.

Tetapkan tipe variabel ke daftar menggunakan konsol Amazon Fraud Detector

Untuk menetapkan jenis variabel ke daftar

1. Luncurkan AWSManagement Console dan masuk ke akun Anda. Luncurkan ke Amazon Fraud Detector.
2. Di panel navigasi sebelah kiri, pilih Daftar.
3. Di halaman Daftar, pilih daftar yang ingin Anda tetapkan tipe variabel.
4. Di halaman detail daftar Anda, pilih Tindakan dan pilih Edit daftar.
5. Di kotak daftar Edit, pilih jenis variabel untuk daftar Anda.
6. Pilih Save (Simpan).

Menetapkan jenis variabel ke daftar menggunakanAWS SDK for Python (Boto3)

Contoh berikut menggunakan operasi UpdateListAPI untuk menetapkan jenis variabel untuk`allow_ip_address` daftar.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.update_list (
     name = 'allow_ip_address',
     variableType = 'IP_ADDRESS'
)
```

Menghapus daftar

Anda dapat menghapus daftar yang tidak digunakan dalam aturan apa pun. Saat Anda menghapus daftar, Amazon Fraud Detector menghapus daftar tersebut dan semua entri dalam daftar secara permanen.

Anda dapat menghapus daftar di konsol Amazon Fraud Detector, menggunakan API, menggunakanAWS CLI atauAWS SDK.

Hapus daftar menggunakan konsol Amazon Fraud Detector

Cara menghapus daftar

1. Luncurkan AWSManagement Console dan masuk ke akun Anda. Luncurkan ke Amazon Fraud Detector.
2. Di panel navigasi sebelah kiri, pilih Daftar
3. Di halaman daftar, pilih daftar yang ingin Anda hapus.
4. Di halaman detail daftar Anda, pilih Tindakan dan pilih Hapus daftar.
5. Pilih Hapus daftar.

Hapus daftar menggunakanAWS SDK for Python (Boto3)

Contoh berikut menggunakan operasi DeleteListAPI untuk menghapus`allow_email_ids`.

```
import boto3
```

```
                        fraudDetector = boto3.client('frauddetector')
fraudDetector.delete_list(
   name = 'allow_email_ids'
)
```

Menghapus entri dari daftar

Anda dapat menghapus satu atau beberapa entri dari daftar Anda kapan saja. Ketika Anda menghapus entri dalam daftar Anda, Anda tidak perlu memperbarui aturan yang terkait dengan daftar. Aturan secara otomatis menggabungkan daftar yang diperbarui.

Anda dapat menghapus entri dari daftar di konsol Amazon Fraud Detector, menggunakan API, menggunakanAWS CLI atauAWS SDK.

Menghapus entri dari daftar menggunakan konsol Amazon Fraud Detector

Untuk menghapus satu atau beberapa entri dari daftar

1. Luncurkan AWSManagement Console dan masuk ke akun Anda. Luncurkan ke Amazon Fraud Detector.
2. Di panel navigasi sebelah kiri, pilih Daftar
3. Di halaman daftar, pilih daftar yang berisi entri yang ingin Anda hapus.
4. Di halaman detail daftar Anda, pilih tab Daftar data dan pilih entri yang ingin Anda hapus.
5. Pilih Hapus dan pilih Hapus lagi untuk mengonfirmasi.

Hapus entri dari daftar menggunakanAWS SDK for Python (Boto3)

Pada contoh berikut operasi UpdateListAPI menghapus entri dari`allow_email_ids` daftar.

```
import boto3
                        fraudDetector = boto3.client('frauddetector')
fraudDetector.update_list(
   name = 'allow_email_ids',
   updateMode = 'REMOVE',
   elements = ['emailId_4', 'emailId_12']
)
```

Hapus semua entri dari daftar

Anda dapat menghapus semua entri dalam daftar Anda, jika daftar tidak digunakan dalam aturan. Anda dapat menghapus semua entri yang ada dalam daftar dan kemudian menambahkan entri dalam daftar yang sama.

Anda dapat menghapus entri dari daftar di konsol Amazon Fraud Detector, menggunakan API, menggunakanAWS CLI atauAWS SDK.

Menghapus semua entri dari daftar menggunakan konsol Amazon Fraud Detector

Untuk menghapus semua entri dari daftar

1. Luncurkan AWSManagement Console dan masuk ke akun Anda. Luncurkan ke Amazon Fraud Detector.
2. Di panel navigasi sebelah kiri, pilih Daftar
3. Di halaman daftar, pilih daftar yang berisi entri yang ingin Anda hapus.
4. Di halaman detail daftar Anda, pilih tab Daftar data dan pilih Hapus semua.
5. Dalam Hapus semua kotak, ketik`delete all` untuk mengonfirmasi lalu pilih Hapus semua data daftar.

Hapus semua entri dari daftar menggunakanAWS SDK for Python (Boto3)

Pada contoh berikut operasi UpdateListAPI menghapus semua entri dari`allow_email_ids` daftar.

```
import boto3
                        fraudDetector = boto3.client('frauddetector')
fraudDetector.update_list(
   name = 'allow_email_ids',
   updateMode = 'REPLACE',
   elements = []
)
```

Hasil

Hasilnya adalah hasil dari prediksi penipuan. Anda dapat membuat hasil untuk setiap kemungkinan hasil prediksi penipuan. Misalnya, Anda mungkin ingin hasil mewakili tingkat risiko (high_risk,

medium_risk, dan low_risk) atau tindakan (approve, review). Setelah hasil dibuat, Anda dapat menambahkan satu atau beberapa hasil. Sebagai bagian dari GetEventPredictionrespons, Amazon Fraud Detector mengembalikan hasil yang ditentukan untuk aturan yang cocok.

Buat hasil

Anda dapat membuat hasil di konsol Amazon Fraud Detector, menggunakan perintah put-outcome, menggunakan PutOutcomeAPI, atau menggunakanAWS SDK for Python (Boto3).

Membuat hasil menggunakan konsol Amazon Fraud Detector

Untuk membuat satu atau beberapa hasil,

1. Buka AWSManagement Console dan masuk ke akun. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi sebelah kiri, pilih Hasil.
3. Di halaman Hasil, pilih Buat.
4. Di halaman hasil baru Anda, masukkan berikut ini:
    a. Dalam nama Hasil, masukkan nama untuk hasil Anda.
    b. Di Deskripsi Hasil, secara opsional, masukkan deskripsi.
5. Pilih Simpan hasil.
6. Ulangi langkah 2 hingga 5 untuk menciptakan hasil tambahan.

Buat hasil menggunakanAWS SDK for Python (Boto3)

Contoh berikut menggunakan`PutOutcome` API untuk membuat tiga hasil. Merekaverify_customer,review, danapprove. Setelah hasil dibuat, Anda dapat menetapkannya ke aturan.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_outcome(
name = 'verify_customer',
description = 'this outcome initiates a verification workflow'
)

fraudDetector.put_outcome(
name = 'review',
```

```
description = 'this outcome sidelines event for review'
)

fraudDetector.put_outcome(
name = 'approve',
description = 'this outcome approves the event'
)
```

Menghapus hasil

Anda tidak dapat menghapus hasil yang digunakan dalam versi aturan.

Saat Anda menghapus hasil, Amazon Fraud Detector menghapus hasil tersebut secara permanen dan data tidak lagi disimpan di Amazon Fraud Detector.

Anda dapat menghapus hasil di konsol Amazon Fraud Detector, menggunakan perintah delete-outcome, menggunakan DeleteOutcomeAPI, atau menggunakanAWS SDK for Python (Boto3)

Menghapus hasil di konsol Amazon Fraud Detector

Untuk menghapus hasil

1. Masuk keAWS Management Console dan buka konsol Amazon Fraud Detector di https://console.aws.amazon.com/frauddetector.
2. Di panel navigasi kiri konsol Amazon Fraud Detector, pilih Sumber Daya, lalu pilih Hasil.
3. Pilih hasil yang ingin Anda hapus.
4. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
5. Masukkan nama hasil, lalu pilih Hapus hasil.

Menghapus hasil menggunakanAWS SDK for Python (Boto3)

Contoh berikut menggunakan DeleteOutcomeAPI untuk menghapus`verify_customer` hasilnya. Setelah hasilnya dihapus, Anda tidak dapat lagi menetapkannya ke aturan.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.delete_outcome(
name = 'verify_customer'
```

```
)
```

Entitas

Entitas mewakili orang atau hal yang melakukan acara tersebut. Jenis entitas mengklasifikasikan entitas. Contoh klasifikasi termasuk pelanggan, atau akun. Anda menyediakan tipe entitas (ENTITY_TYPE) dan pengenal entitas (ENTITY_ID) sebagai bagian dari kumpulan data peristiwa Anda untuk menunjukkan entitas tertentu yang melakukan peristiwa tersebut.

Amazon Fraud Detector menggunakan jenis entitas saat membuat prediksi penipuan untuk suatu peristiwa untuk menunjukkan siapa yang melakukan peristiwa tersebut. Jenis entitas yang ingin Anda gunakan dalam prediksi penipuan Anda harus terlebih dahulu dibuat di Amazon Fraud Detector dan kemudian ditambahkan ke acara saat membuat jenis peristiwa Anda.

Buat jenis

Anda dapat membuat jenis entitas di konsol Amazon Fraud Detector, menggunakan put-entity-typeperintah, menggunakan PutEntityTypeAPI, atau menggunakanAWS SDK for Python (Boto3). Contoh di bawah ini membuat jenis`customer` di konsol Amazon Fraud Detector dan menggunakan SDK for Python (Boto3). Jika Anda membuat jenis entitas untuk dikaitkan dengan jenis peristiwa untuk melatih model deteksi penipuan, gunakan jenis entitas dari kumpulan data peristiwa yang sesuai untuk kasus penggunaan Anda.

Membuat jenis entitas menggunakan konsol Amazon Fraud Detector

Untuk membuat jenis,

1. AWSLanjutkan ke akun Anda.
2. Arahkan ke Amazon Fraud Detector, pilih Entitas di navigasi kiri, lalu pilih Buat.
3. Di halaman Buat entitas, masukkan pelanggan sebagai nama jenis entitas. Atau, masukkan deskripsi.
4. Pilih Buat entitas.

Membuat jenis entitas menggunakanAWS SDK for Python (Boto3)

ContohAWS SDK for Python (Boto3) kode berikut menggunakan`PutEntityType` API untuk membuat jenis entitas`customer`. Jika Anda membuat jenis entitas untuk dikaitkan dengan jenis

peristiwa untuk melatih model deteksi penipuan, gunakan entitas dari kumpulan data peristiwa yang sesuai untuk kasus penggunaan Anda.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.put_entity_type(
name = 'customer',
description = 'customer'
)
```

Hapus jenis

Di Amazon Fraud Detector, Anda tidak dapat menghapus jenis entitas yang disertakan dalam jenis peristiwa. Anda harus terlebih dahulu menghapus jenis acara entitas dikaitkan dengan dan kemudian menghapus jenis entitas.

Saat Anda menghapus jenis entitas, Amazon Fraud Detector menghapus jenis entitas tersebut secara permanen dan data tidak lagi disimpan di Amazon Fraud Detector.

Jenis entitas dapat dihapus di konsol Amazon Fraud Detector, menggunakan delete-entity-typeperintah, menggunakan DeleteEntityTypeAPI, atau menggunakanAWS SDK for Python (Boto3)

Menghapus jenis entitas di konsol Amazon Fraud Detector

Untuk menghapus jenis,

1. Masuk keAWS Management Console dan buka Fraud Detector https://console.aws.amazon.com/frauddetector.
2. Di panel navigasi kiri konsol Amazon Fraud Detector, lalu pilih Enitas.
3. Pilih jenis yang ingin Anda hapus.
4. Pilih Actions (Tindakan), lalu pilih Delete (Hapus).
5. Masukkan nama jenis entitas, dan kemudian pilih Hapus jenis entitas.

Hapus jenis entitas menggunakanAWS SDK for Python (Boto3)

AWS SDK for Python (Boto3)Contoh kode berikut menghapus pelanggan jenis entitas menggunakan DeleteEntityTypeAPI.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.delete_entity_type (

name = 'customer'

)
```

Mengelola sumber daya Amazon Fraud Detector menggunakanAWS CloudFormation

Amazon Fraud Detector Fraud DetectorAWS CloudFormation Anda membuat templat yang menggambarkan semua sumber daya Amazon Fraud Detector EntityType EventTypeAWS CloudFormation Anda dapat menggunakan kembali template untuk menyediakan dan mengonfigurasi sumber daya secara konsisten dan berulang kali di beberapa akun dan Wilayah AWS.

Tidak ada biaya tambahan untuk menggunakan AWS CloudFormation.

Templat templat Amazon Fraud Detector

Untuk menyediakan dan mengonfigurasi sumber daya untuk Amazon Fraud DetectorAWS CloudFormation Templat adalah file teks dengan format JSON atau YAML. Templat ini menjelaskan sumber daya yang ingin Anda sediakan di tumpukan AWS CloudFormation Anda. Jika Anda tidak terbiasa dengan JSON atau YAML, Anda dapat menggunakan AWS CloudFormation Designer untuk membantu Anda memulai dengan templat AWS CloudFormation. Untuk informasi selengkapnya, lihat Apa yang dimaksud dengan AWS CloudFormation Designer? dalam Panduan Pengguna AWS CloudFormation.

Anda juga dapat membuat, memperbarui, memperbarui, menghapus sumber daya Amazon Fraud DetectorAWS CloudFormation Untuk informasi selengkapnya, termasuk contoh templat JSON dan YAKL untuk sumber daya Anda, lihat referensi tipe sumber daya Amazon Fraud DetectorAWS CloudFormation

Jika Anda sudah menggunakan CloudFormation, tidak perlu mengelola kebijakan IAM tambahan atau CloudTrail logging.

Mengelola Amazon Fraud Detector

Anda dapat membuat, memperbarui, dan menghapus tumpukan Amazon Fraud Detector melalui CloudFormation konsol atau melalui AWS CLI.

Untuk membuat tumpukan, Anda harus memiliki templat yang menggambarkan sumber daya apa yang CloudFormation akan disertakan AWS dalam tumpukan Anda. Anda juga dapat membawa sumber daya Amazon Fraud Detector yang telah Anda buat ke dalam CloudFormation manajemen dengan mengimpornya ke tumpukan baru atau yang sudah ada.

Untuk petunjuk mendetail untuk mengelola tumpukan Anda, lihat PanduanAWS CloudFormation Pengguna untuk mempelajari cara membuat, memperbarui, dan menghapus tumpukan.

Mengelola Amazon Fraud Detector

Cara Anda mengaturAWS CloudFormation tumpukan Anda sepenuhnya terserah Anda. Hal ini umumnya praktek terbaik adalah untuk mengatur tumpukan oleh siklus hidup dan kepemilikan. Ini berarti mengelompokkan sumber daya berdasarkan seberapa sering mereka berubah atau oleh tim yang bertanggung jawab untuk memperbaruinya.

Anda dapat memilih untuk mengatur tumpukan Anda dengan membuat tumpukan untuk setiap detektor dan logika pendeteksiannya (misalnya, aturan, variabel, dll.). Jika Anda menggunakan layanan lain, Anda harus mempertimbangkan apakah Anda ingin mengumpulkan sumber daya Amazon Fraud Detector dengan sumber daya dari layanan lain. Misalnya, Anda dapat membuat tumpukan yang menyertakan sumber daya Kinesis yang membantu mengumpulkan data dan sumber daya Amazon Fraud Detector yang memproses data. Ini bisa menjadi cara yang efektif untuk memastikan bahwa semua produk tim penipuan Anda bekerja sama.

Memahami CloudFormation parameter Amazon Fraud Detector

Selain parameter standar yang tersedia di semua CloudFormation template, Amazon Fraud Detector memperkenalkan dua parameter tambahan yang akan membantu Anda mengelola perilaku penerapan. Jika Anda tidak menyertakan salah satu atau kedua parameter ini, CloudFormation akan menggunakan nilai default yang ditunjukkan di bawah ini.

Parameter	Nilai	nilai default
DetectorVersionStatus	AKTIF: Atur versi detektor baru/diperbarui ke status Aktif	DRAF

Parameter	Nilai	nilai default
	DRAFT: Atur versi detektor baru/diperbarui ke status Draft	
Inline	TRUE: Izinkan CloudFormation untuk membuat/memperbarui/menghapus sumber daya saat membuat/memperbarui/menghapus tumpukan. FALSE: Izinkan CloudFormation untuk memvalidasi bahwa objek ada tetapi tidak membuat perubahan pada objek.	BETUL

ContohAWS CloudFormation template untuk sumber daya Amazon Fraud Detector

Berikut ini adalah contoh templatAWS CloudFormation YAKL untuk mengelola detector dan versi detector terkait.

```
# Simple Detector resource containing inline Rule, EventType, Variable, EntityType and
 Label resource definitions
Resources:
  TestDetectorLogicalId:
    Type: AWS::FraudDetector::Detector
    Properties:
      DetectorId: "sample_cfn_created_detector"
      DetectorVersionStatus: "DRAFT"
      Description: "A detector defined and created in a CloudFormation stack!"

      Rules:
        - RuleId: "over_threshold_investigate"
          Description: "Automatically sends transactions of $10000 or more to an
 investigation queue"
          DetectorId: "sample_cfn_created_detector"
          Expression: "$amount >= 10000"
          Language: "DETECTORPL"
          Outcomes:
            - Name: "investigate"
              Inline: true
```

```
      - RuleId: "under_threshold_approve"
        Description: "Automatically approves transactions of less than $10000"
        DetectorId: "sample_cfn_created_detector"
        Expression: "$amount <10000"
        Language: "DETECTORPL"
        Outcomes:
          - Name: "approve"
            Inline: true
    EventType:
      Inline: "true"
      Name: "online_transaction"
      EventVariables:
        - Name: "amount"
          DataSource: 'EVENT'
          DataType: 'FLOAT'
          DefaultValue: '0'
          VariableType: "PRICE"
          Inline: 'true'
      EntityTypes:
        - Name: "customer"
          Inline: 'true'
      Labels:
        - Name: "legitimate"
          Inline: 'true'
        - Name: "fraudulent"
          Inline: 'true'
```

Pelajari selengkapnya tentang AWS CloudFormation

Untuk mempelajari selengkapnya tentang AWS CloudFormation, lihat sumber daya berikut:

- AWS CloudFormation
- AWS CloudFormationPanduan Pengguna
- AWS CloudFormationReferensi API
- AWS CloudFormationPanduan Pengguna Bar

Prediksi penipuan

Anda dapat menggunakan Amazon Fraud Detector untuk mendapatkan prediksi penipuan untuk acara tunggal secara real time atau mendapatkan prediksi penipuan secara offline untuk serangkaian acara. Untuk menghasilkan prediksi penipuan untuk satu peristiwa atau serangkaian peristiwa, Anda harus memberikan informasi berikut kepada Amazon Fraud Detector:

- Logika prediksi penipuan
- Metadata peristiwa

Logika deteksi penipuan

Logika prediksi penipuan menggunakan satu atau lebih aturan untuk mengevaluasi data yang terkait dengan suatu peristiwa dan kemudian memberikan hasil dan skor prediksi penipuan. Anda membuat logika prediksi penipuan Anda dengan menggunakan komponen berikut:

- Jenis acara - Mendefinisikan struktur acara
- Model - Mendefinisikan algoritma dan persyaratan data untuk memprediksi penipuan
- Variabel - Merupakan elemen data yang terkait dengan acara
- Aturan - Memberitahu Amazon Fraud Detector mengenai cara menafsirkan nilai variabel selama prediksi penipuan
- Hasil - Hasil yang dihasilkan dari prediksi penipuan
- Versi detektor - Berisi logika prediksi penipuan untuk acara tertentu

Untuk informasi selengkapnya tentang komponen yang digunakan untuk membuat logika deteksi penipuan, lihat konsep Amazon Fraud Detector. Sebelum Anda mulai membuat prediksi penipuan, pastikan Anda telah membuat dan menerbitkan versi detektor yang berisi logika prediksi penipuan Anda. Anda dapat membuat dan mempublikasikan versi detektor menggunakan Konsol Fraud Detector atau API. Untuk petunjuk tentang penggunaan konsol, lihat Memulai (konsol). Untuk petunjuk penggunaan API, lihat Membuat versi detektor.

Metadata peristiwa

Metadata peristiwa memberikan rincian peristiwa yang sedang dievaluasi. Setiap peristiwa yang ingin Anda evaluasi harus menyertakan nilai untuk setiap variabel dalam jenis peristiwa yang terkait dengan versi detektor Anda. Selain itu, metadata peristiwa harus menyertakan hal berikut:

- EVENT_ID - Pengidentifikasi peristiwa tersebut. Misalnya, jika acara Anda adalah transaksi online, EVENT_ID mungkin merupakan nomor referensi transaksi yang diberikan kepada pelanggan Anda.

  Catatan penting tentang EVENT_ID

  - Harus bersifat unik untuk peristiwa tersebut
  - Harus mewakili informasi yang berarti bagi bisnis Anda
  - Harus memenuhi pola ekspresi reguler:`^[0-9a-z_-]+$.`
  - Harus diselamatkan. EVENT_ID adalah referensi untuk acara tersebut dan digunakan untuk melakukan operasi pada acara seperti menghapus acara tersebut.
  - Menambahkan stempel waktu ke EVENT_ID tidak disarankan karena dapat menyebabkan masalah saat nanti Anda ingin memperbarui acara, karena Anda harus memberikan EVENT_ID yang sama persis.
- ENTITY_TYPE — Entitas yang melakukan acara, seperti pedagang atau pelanggan.
- ENTITY_ID - Sebuah identifier untuk entitas melakukan acara tersebut. ENTITY_ID harus memenuhi pola ekspresi reguler berikut:`^[0-9a-z_-]+$`. Jika ENTITY_ID tidak tersedia pada saat evaluasi, lulus string yang tidak diketahui.
- EVENT_TIMESTAMP - The timestamp ketika peristiwa terjadi. Cap waktu harus dalam standar ISO 8601 di UTC.

Prediksi waktu nyata

Anda dapat mengevaluasi aktivitas online untuk penipuan secara real time dengan meneleponGetEventPrediction API. Anda memberikan informasi tentang satu peristiwa dalam setiap permintaan dan secara sinkron menerima skor model dan hasil berdasarkan logika prediksi penipuan yang terkait dengan detektor yang ditentukan.

Cara kerja prediksi penipuan waktu nyata

GetEventPredictionAPI menggunakan versi detektor tertentu untuk mengevaluasi metadata peristiwa yang disediakan untuk acara tersebut. Selama evaluasi, Amazon Fraud Detector pertama-tama menghasilkan skor model untuk model yang ditambahkan ke versi detektor, kemudian meneruskan hasilnya ke aturan untuk evaluasi. Aturan dijalankan seperti yang ditentukan oleh mode eksekusi aturan (lihat Membuat versi detektor). Sebagai bagian dari respons, Amazon Fraud Detector memberikan skor model serta hasil apa pun yang terkait dengan aturan yang cocok.

Mendapatkan prediksi penipuan waktu nyata

Untuk mendapatkan prediksi penipuan secara real time, pastikan Anda telah membuat dan menerbitkan detektor yang berisi model dan aturan prediksi penipuan Anda, atau sekadar kumpulan aturan.

Anda bisa mendapatkan prediksi penipuan untuk suatu peristiwa secara real time dengan memanggil operasi GetEventPredictionAPI menggunakanAWS Command Line Interface (AWSCLI) atau salah satu SDK Amazon Fraud Detector.

Untuk menggunakan API, berikan informasi dari satu peristiwa dengan setiap permintaan. Sebagai bagian dari permintaan, Anda harus menentukan`detectorId` bahwa Amazon Fraud Detector akan digunakan untuk mengevaluasi kejadian. Anda dapat secara opsional menentukan`detectorVersionId`. Jika tidak`detectorVersionId` ditentukan, Amazon Fraud Detector akan menggunakan`ACTIVE` versi detektor.

Anda opsional dapat mengirim data untuk memanggil SageMaker model dengan melewati data di lapangan`externalModelEndpointBlobs`.

Dapatkan prediksi penipuan menggunakanAWS SDK for Python (Boto3)

Untuk menghasilkan prediksi penipuan, panggil`GetEventPrediction` API. Contoh di bawah ini mengasumsikan Anda telah menyelesaikanBagian B: Menghasilkan prediksi penipuan. Sebagai bagian dari respons, Anda akan menerima skor model serta aturan yang cocok dan hasil yang sesuai. Anda dapat menemukan contoh`GetEventPrediction` permintaan tambahan pada aws-fraud-detector-samples GitHub repositori.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.get_event_prediction(
detectorId = 'sample_detector',
eventId = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
eventTypeName = 'sample_registration',
eventTimestamp = '2020-07-13T23:18:21Z',
entities = [{'entityType':'sample_customer', 'entityId':'12345'}],
eventVariables = {
    'email_address' : 'johndoe@exampledomain.com',
    'ip_address' : '1.2.3.4'
}
```

```
)
```

Prediksi Batch

Anda dapat menggunakan pekerjaan prediksi batch di Amazon Fraud Detector untuk mendapatkan prediksi untuk serangkaian peristiwa yang tidak memerlukan penilaian waktu nyata. Misalnya, Anda dapat membuat pekerjaan prediksi batch untuk melakukan offlineproof-of-concept, atau untuk secara retrospektif mengevaluasi risiko kejadian setiap jam, harian, atau mingguan.

Anda dapat membuat tugas prediksi batch menggunakan konsol Amazon Fraud Detector, atau dengan memanggil operasi CreateBatchPredictionJobAPI menggunakan AWS Command Line Interface (AWSCLI) atau salah satu SDK Amazon Fraud Detector.

Topik

- Cara kerja prediksi batch
- File input dan output
- Mendapatkan prediksi batch
- Panduan tentang Peran IAM
- Dapatkan prediksi penipuan batch menggunakan AWS SDK for Python (Boto3)

Cara kerja prediksi batch

Operasi `CreateBatchPredictionJob` API menggunakan versi detektor tertentu untuk membuat prediksi berdasarkan data yang disediakan dalam file CSV masukan yang terletak di bucket Amazon S3. API kemudian mengembalikan file CSV yang dihasilkan ke bucket S3.

Pekerjaan prediksi Batch menghitung skor model dan hasil prediksi dengan cara yang sama seperti operasi. `GetEventPrediction` Mirip dengan`GetEventPrediction`, untuk membuat pekerjaan prediksi batch, Anda pertama kali membuat jenis peristiwa, opsional melatih model, dan kemudian membuat versi detektor yang mengevaluasi peristiwa dalam pekerjaan batch Anda.

Harga untuk skor risiko peristiwa yang dievaluasi oleh pekerjaan prediksi batch sama dengan harga untuk skor yang dibuat oleh API. `GetEventPrediction` Untuk detailnya, lihat harga Amazon Fraud Detector.

Anda hanya dapat menjalankan satu batch prediksi job dalam satu waktu.

File input dan output

File CSV masukan harus berisi header yang cocok dengan jenis peristiwa yang dikaitkan dengan versi detektor yang dipilih. Ukuran maksimum file data input adalah 1GB. Jumlah acara akan bervariasi menurut ukuran acara Anda.

Amazon Fraud Detector membuat file keluaran dalam bucket yang sama dengan file input, kecuali Anda menentukan lokasi terpisah untuk data keluaran. File output berisi data asli dari file input dan kolom ditambahkan berikut:

- `MODEL_SCORES`- Detail skor model untuk acara dari setiap model yang terkait dengan versi detektor yang dipilih.
- `OUTCOMES`- Merinci hasil peristiwa seperti yang dievaluasi oleh versi detektor yang dipilih dan aturannya.
- `STATUS`- Menunjukkan apakah acara tersebut berhasil dievaluasi. Jika acara tidak berhasil dievaluasi, kolom ini menunjukkan kode alasan kegagalan.
- `RULE_RESULTS`- Daftar semua aturan yang cocok, berdasarkan mode eksekusi aturan.

Mendapatkan prediksi batch

Langkah-langkah berikut mengasumsikan bahwa Anda telah membuat jenis peristiwa, melatih model menggunakan jenis peristiwa tersebut (opsional), dan membuat versi detektor untuk jenis peristiwa tersebut.

Untuk mendapatkan prediksi batch

1. Masuklah ke AWS Management Console dan buka konsol Amazon Fraud Detector di https://console.aws.amazon.com/frauddetector.
2. Di panel navigasi kiri konsol Amazon Fraud Detector, pilih Prediksi Batch, lalu pilih Prediksi batch baru.
3. Di Nama Job, tentukan nama untuk pekerjaan prediksi batch Anda. Jika Anda tidak menentukan nama, Amazon Fraud Detector akan menghasilkan nama pekerjaan secara acak.
4. Di Detektor, pilih detektor untuk prediksi batch ini.
5. Dalam versi Detektor, pilih versi detektor untuk prediksi batch ini. Anda dapat memilih versi detektor dalam status apa pun. Jika detektor Anda memiliki versi detektor dalam `Active`

status, versi tersebut dipilih secara otomatis, tetapi Anda juga dapat mengubah pilihan ini jika diperlukan.

6. Dalam peran IAM, pilih atau buat peran yang memiliki akses baca dan tulis ke bucket Amazon S3 input dan output Anda. Lihat Panduan tentang Peran IAM untuk informasi selengkapnya.

   Untuk mendapatkan prediksi batch, peran IAM yang memanggil `CreateBatchPredictionJob` operasi harus memiliki izin baca ke bucket S3 input dan izin tulis ke bucket S3 keluaran Anda. Untuk informasi selengkapnya tentang izin bucket, lihat Contoh kebijakan pengguna di Panduan Pengguna Amazon S3.
7. Di Lokasi data input, tentukan lokasi Amazon S3 data input. Jika Anda menginginkan file keluaran dalam bucket S3 yang berbeda, pilih Pisahkan lokasi data untuk output dan sediakan lokasi Amazon S3 untuk data keluaran Anda.
8. (Opsional) Buat tag untuk pekerjaan prediksi batch Anda.
9. Pilih Mulai.

   Amazon Fraud Detector menciptakan pekerjaan prediksi batch, dan status pekerjaan adalah. `In progress` Waktu pemrosesan pekerjaan prediksi Batch bervariasi tergantung pada jumlah kejadian dan konfigurasi versi detektor Anda.

Untuk menghentikan pekerjaan prediksi batch yang sedang berlangsung, buka halaman detail pekerjaan prediksi batch, pilih Tindakan, lalu pilih Hentikan prediksi batch. Jika Anda menghentikan pekerjaan prediksi batch, Anda tidak akan menerima hasil apa pun untuk pekerjaan itu.

Ketika status pekerjaan prediksi batch berubah`Complete`, Anda dapat mengambil output pekerjaan dari bucket Amazon S3 keluaran yang ditentukan. Nama file output dalam format`batch prediction job name_file creation timestamp_output.csv`. Misalnya, file output dari pekerjaan bernama `mybatchjob` adalah`mybatchjob_ 1611170650_output.csv`.

Untuk mencari peristiwa tertentu yang dievaluasi oleh tugas prediksi batch, di panel navigasi kiri konsol Amazon Fraud Detector, pilih Cari prediksi sebelumnya.

Untuk menghapus tugas prediksi batch yang telah selesai, buka halaman detail pekerjaan prediksi batch, pilih Tindakan, lalu pilih Hapus prediksi batch.

Panduan tentang Peran IAM

Untuk mendapatkan prediksi batch, peran IAM yang memanggil CreateBatchPredictionJoboperasi harus memiliki izin baca ke bucket S3 input dan izin tulis ke bucket S3 keluaran Anda. Untuk

informasi selengkapnya tentang izin bucket, lihat Contoh kebijakan pengguna di Panduan Pengguna Amazon S3. Di konsol Amazon Fraud Detector, Anda memiliki tiga opsi untuk memilih peran IAM untuk Prediksi Batch:

1. Buat peran saat membuat pekerjaan Prediksi Batch baru.
2. Pilih peran IAM yang sudah ada yang telah Anda buat sebelumnya di konsol Amazon Fraud Detector. Pastikan untuk menambahkan `S3:PutObject` izin ke peran sebelum Anda melakukan langkah ini.
3. Masukkan ARN kustom untuk peran IAM yang dibuat sebelumnya.

Jika Anda menerima kesalahan terkait dengan Peran IAM, verifikasi hal berikut:

1. Bucket input dan output Amazon S3 berada di wilayah yang sama dengan detektor.
2. Peran IAM yang Anda gunakan memiliki `s3:GetObject` izin untuk bucket S3 input Anda dan `s3:PutObject` izin untuk bucket S3 keluaran Anda.
3. Peran IAM yang Anda gunakan memiliki kebijakan kepercayaan untuk prinsipal `frauddetector.amazonaws.com` layanan.

Dapatkan prediksi penipuan batch menggunakan AWS SDK for Python (Boto3)

Contoh berikut menunjukkan permintaan sampel untuk CreateBatchPredictionJobAPI. Pekerjaan prediksi batch harus menyertakan sumber daya berikut yang ada: detektor, versi detektor, dan nama jenis peristiwa. Contoh berikut mengasumsikan Anda telah membuat jenis peristiwa`sample_registration`, detektor`sample_detector`, dan versi 1 detektor.

```
import boto3
fraudDetector = boto3.client('frauddetector')

fraudDetector.create_batch_prediction_job (
   jobId = 'sample_batch',
   inputPath = 's3://bucket_name/input_file_name.csv',
   outputPath = 's3://bucket_name/',
   eventTypeName = 'sample_registration',
   detectorName = 'sample_detector',
   detectorVersion = '1',
   iamRoleArn = 'arn:aws:iam::**:role/service-role/AmazonFraudDetector-DataAccessRole-
**'
```

```
)
```

Penjelasan prediksi

Penjelasan prediksi memberikan wawasan tentang bagaimana setiap variabel peristiwa memengaruhi skor prediksi penipuan model Anda, dan secara otomatis dihasilkan sebagai bagian dari prediksi penipuan. Setiap prediksi penipuan dilengkapi dengan skor risiko antara 1 dan 1000. Penjelasan prediksi memberi Anda rincian pengaruh setiap variabel peristiwa pada skor risiko dalam hal besarnya (0-5, 5 tertinggi) dan arah (skor mengemudi lebih tinggi atau lebih rendah). Anda juga dapat menggunakan penjelasan prediksi untuk tugas-tugas berikut:

- Untuk mengidentifikasi indikator risiko teratas selama inverstigasi manual ketika suatu peristiwa ditandai untuk ditinjau.
- Untuk mempersempit akar penyebab yang mengarah pada prediksi positif palsu (misalnya, skor risiko tinggi untuk peristiwa yang sah).
- Untuk menganalisis pola penipuan di seluruh data peristiwa dan mendeteksi bias, jika ada, dalam kumpulan data Anda.

> **Important**
>
> Penjelasan prediksi dibuat secara otomatis dan hanya tersedia untuk model yang dilatih pada atau setelah 30 Juni 2021. Untuk menerima penjelasan prediksi untuk model yang dilatih sebelum 30 Juni 2021, latih kembali model-model tersebut.

Penjelasan prediksi memberikan serangkaian nilai berikut untuk setiap variabel peristiwa yang digunakan untuk melatih model.

Dampak relatif

Memberikan referensi visual tentang dampak variabel dalam hal besarnya pada skor prediksi penipuan. Nilai dampak relatif terdiri dari peringkat bintang (0-5, 5 menjadi yang tertinggi) dan arah (meningkat/menurun) dampak risiko penipuan.

- Variabel yang meningkatkan risiko penipuan ditunjukkan oleh bintang berwarna merah. Semakin tinggi jumlah bintang berwarna merah, semakin banyak variabel yang mendorong skor penipuan dan meningkatkan kemungkinan penipuan.

- Variabel yang menurunkan risiko penipuan ditunjukkan oleh bintang berwarna hijau. Semakin tinggi jumlah awal berwarna hijau, semakin banyak variabel yang menurunkan skor risiko penipuan dan penurunan kemungkinan penipuan.
- Bintang nol untuk semua variabel menunjukkan bahwa tidak ada variabel sendiri yang secara signifikan mengubah risiko penipuan.

Nilai penjelasan mentah

Memberikan nilai mentah dan tidak ditafsirkan yang direpresentasikan sebagai log-odds penipuan. Nilai-nilai ini biasanya antara -10 hingga +10, tetapi berkisar dari - tak terhingga hingga + tak terhingga.

- Nilai positif menunjukkan bahwa variabel mendorong skor risiko naik.
- Nilai negatif menunjukkan bahwa variabel mendorong skor risiko turun.

Di konsol Amazon Fraud Detector, nilai penjelasan prediksi ditampilkan sebagai berikut. Peringkat bintang berwarna dan nilai numerik mentah yang sesuai memudahkan untuk melihat pengaruh relatif antar variabel.

Prediction explanations - *preview*

This prediction is based on contribution from each variable to the overall likelihood of a fraudulent event. Prediction explanations give you better understanding of how an event's input variables influence fraud prediction scores. For details on calculations, refer to documentation

Show raw prediction explanation value

Variables that increased fraud risk

Name	Value	Relative impact	Raw explanation value
comp_255	whatsapp	★★★★★	0.49
req_255	0	★☆☆☆☆	0.29
sentiment_description	0.2	★☆☆☆☆	0.12
desc_255	this is the company description	☆☆☆☆☆	0.07
title	king	☆☆☆☆☆	0.07
required_experience	5	☆☆☆☆☆	0.04
required_education	masters	☆☆☆☆☆	0.03
has_questions	true	☆☆☆☆☆	0.01

Variables that decreased fraud risk

Name	Value	Relative impact	Raw explanation value
has_company_logo	true	★★★★★	-0.26
req_desc_similarity	0.3	★☆☆☆☆	-0.21
employment_type	temp	★☆☆☆☆	-0.21
job_location	california	☆☆☆☆☆	-0.11
job_function	engineer	☆☆☆☆☆	-0.06
industry	software	☆☆☆☆☆	-0.05
sentiment_requirements	0.5	☆☆☆☆☆	-0.01
telecommuting	yes	☆☆☆☆☆	-0.00
company_desc_similarity	0.0	☆☆☆☆☆	-0.00

Melihat penjelasan prediksi

Setelah menghasilkan prediksi penipuan, Anda dapat melihat penjelasan prediksi di konsol Amazon Fraud Detector. Untuk melihat penjelasan prediksi menggunakan API dari AWS SDK, Anda harus terlebih dahulu memanggil `ListEventPrediction` API untuk mendapatkan stempel waktu prediksi untuk acara tersebut, lalu memanggil `GetEventPredictionMetadata` API untuk mendapatkan penjelasan prediksi.

Lihat penjelasan prediksi menggunakan konsol Amazon Fraud Detector

Untuk melihat penjelasan prediksi menggunakan konsol,

1. Buka AWS Konsol dan masuk ke akun Anda. Arahkan ke Amazon Fraud Detector.
2. Di panel navigasi kiri, pilih Cari prediksi masa lalu.
3. Gunakan filter Properti, Operator, dan Nilai untuk memilih prediksi yang ingin Anda tinjau.

4. Di panel Filter atas, pastikan untuk memilih periode waktu kapan prediksi yang ingin Anda tinjau dibuat.
5. Panel Hasil menampilkan daftar semua prediksi yang dihasilkan selama periode waktu yang ditentukan. Klik ID Peristiwa prediksi untuk melihat penjelasan prediksi.
6. Gulir ke bawah ke panel Penjelasan prediksi.
7. Tetapkan tombol Tampilkan nilai penjelasan prediksi mentah untuk melihat nilai penjelasan prediksi mentah dari semua variabel.

Lihat penjelasan prediksi menggunakan AWS SDK for Python (Boto3)

Contoh berikut menunjukkan contoh permintaan untuk melihat penjelasan prediksi menggunakan `ListEventPredictions` dan `GetEventPredictionMetadata` API dari SDKAWS.

Contoh 1: Dapatkan daftar prediksi terbaru menggunakan API **`ListEventPredictions`**

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraudDetector.list_event_predictions(
  maxResults = 10,
  predictionTimeRange = {
     end_time: '2022-01-13T23:18:21Z',
     start_time: '2022-01-13T20:18:21Z'
    }
 )
```

Contoh 2; Dapatkan daftar prediksi sebelumnya untuk jenis acara “registrasi” menggunakan API **`ListEventPredictions`**

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraudDetector.list_event_predictions(
   eventType = {
      value = 'registration'
    }
   maxResults = 70,
   nextToken = "10",
   predictionTimeRange = {
     end_time: '2021-07-13T23:18:21Z',
     start_time: '2021-07-13T20:18:21Z'
```

```
    }
)
```

Contoh 3: Dapatkan detail prediksi sebelumnya untuk ID peristiwa tertentu, jenis peristiwa, ID detektor, dan ID versi detektor yang dihasilkan dalam periode waktu tertentu menggunakan **`GetEventPredictionMetadata`** API.

Yang `predictionTimestamp` ditentukan untuk permintaan ini diperoleh dengan terlebih dahulu memanggil `ListEventPredictions` API.

```
import boto3
fraudDetector = boto3.client('frauddetector')
fraudDetector.get_event_prediction_metadata (
   detectorId = 'sample_detector',
   detectorVersionId = '1',
   eventId = '802454d3-f7d8-482d-97e8-c4b6db9a0428',
   eventTypeName = 'sample_registration',
   predictionTimestamp = '2021-07-13T21:18:21Z'
 )
```

Memahami bagaimana penjelasan prediksi dihitung

Amazon Fraud Detector menggunakan SHAP (ShapeLey Additive Explanations) untuk menjelaskan prediksi peristiwa individu dengan menghitung nilai penjelasan mentah dari setiap variabel peristiwa yang digunakan untuk pelatihan model. Nilai penjelasan mentah dihitung oleh model sebagai bagian dari algoritma klasifikasi saat menghasilkan prediksi. Nilai penjelasan mentah ini mewakili kontribusi setiap masukan terhadap logaritma kemungkinan penipuan. Nilai penjelasan mentah (dari -infinity hingga+infinity) diubah menjadi nilai dampak relatif (-5 hingga +5) menggunakan pemetaan. Nilai dampak relatif yang berasal dari nilai penjelasan mentah mewakili berapa kali peningkatan peluang penipuan (positif) atau legit (negatif), sehingga lebih mudah untuk memahami penjelasan prediksi.

Keamanan di Amazon Fraud Detector

Keamanan cloud di AWS merupakan prioritas tertinggi. Sebagai pelanggan AWS, Anda mendapatkan manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan dari organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara AWS dan Anda. Model tanggung jawab bersama menggambarkan ini sebagai keamanan dari cloud dan keamanan di dalam cloud:

- Keamanan cloud – AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan layanan–layanan AWS di dalam AWS Cloud. AWS juga memberikan Anda layanan yang dapat digunakan dengan aman. Auditor pihak ketiga melakukan pengujian dan verifikasi secara berkala terhadap efektivitas keamanan kami sebagai bagian dari Program Kepatuhan AWS . Untuk mempelajari tentang program kepatuhan yang berlaku untuk Amazon Fraud Detector, lihat AWS Services in Scope by Compliance Program .
- Keamanan di cloud – Tanggung jawab Anda ditentukan menurut layanan AWS yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain, yang mencakup sensitivitas data Anda, persyaratan perusahaan Anda, serta undang-undang dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan Amazon Fraud Detector. Topik berikut menunjukkan cara mengonfigurasi Amazon Fraud Detector untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan sumber daya Amazon Fraud Detector Anda.

Topik

Perlindungan Data di Amazon Fraud Detector

Model tanggung jawab AWS bersama model berlaku untuk perlindungan data di Amazon Fraud Detector. Sebagaimana diuraikan dalam model ini, AWS bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk memelihara kendali atas isi yang dihost pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS.

Untuk tujuan perlindungan data, sebaiknya lindungi kredensial Akun AWS dan siapkan untuk masing-masing pengguna 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 pengelogan aktivitas pengguna dengan AWS CloudTrail.
- Gunakan solusi enkripsi AWS, bersama semua kontrol keamanan bawaan dalam 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 ketika 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 Amazon Fraud Detector atau lainnya Layanan AWS menggunakan konsol, APIAWS CLI, atau AWS SDK. 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.

Mengenkripsi data saat istirahat

Amazon Fraud Detector mengenkripsi data Anda saat istirahat dengan pilihan kunci enkripsi Anda. Anda dapat memilih salah satu dari yang berikut ini:

- Kunci KMS yang AWS dimiliki. Jika Anda tidak menentukan kunci enkripsi, data Anda akan dienkripsi dengan kunci ini secara default.
- Kunci KMS yang dikelola pelanggan. Anda dapat mengontrol akses ke kunci KMS yang dikelola pelanggan menggunakan kebijakan utama. Untuk informasi tentang membuat dan mengelola kunci KMS yang dikelola pelanggan, lihatManajemen kunci.

Mengenkripsi data dalam perjalanan

Amazon Fraud Detector menyalin data dari akun Anda dan memprosesnya dalam AWS sistem internal. Secara default, Amazon Fraud Detector menggunakan TLS 1.2 dengan AWS sertifikat untuk mengenkripsi data dalam perjalanan.

Manajemen kunci

Amazon Fraud Detector mengenkripsi data Anda menggunakan salah satu dari dua jenis kunci:

- Kunci KMS yang AWS dimiliki. Ini menjadi opsi default.
- Kunci KMS yang dikelola pelanggan.

Membuat kunci KMS yang dikelola pelanggan

Anda dapat membuat kunci KMS yang dikelola pelanggan menggunakan konsol AWS KMS atau API. CreateKey Saat membuat kunci pastikan Anda,

- Pilih kunci KMS enkripsi simetris yang dikelola pelanggan, Amazon Fraud Detector tidak mendukung kunci KMS asimetris. Untuk informasi selengkapnya, lihat Kunci Asimetris AWS KMS di Panduan Pengembang Layanan Manajemen AWS Kunci.
- Buat kunci KMS wilayah tunggal. Amazon Fraud Detector tidak mendukung kunci KMS multi-wilayah. Untuk informasi selengkapnya, lihat Kunci multi-wilayah AWS KMS di Panduan Pengembang Layanan Manajemen AWS Kunci.
- Berikan kebijakan utama berikut untuk memberikan izin kepada Amazon Fraud Detector untuk menggunakan kunci tersebut.

```
        {
    "Effect": "Allow",
    "Principal": {
        "Service": "frauddetector.amazonaws.com"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey",
        "kms:CreateGrant",
        "kms:RetireGrant"
    ],
    "Resource": "*"
}
```

Untuk informasi tentang kebijakan utama, lihat Menggunakan Kebijakan Utama di AWS KMS di Panduan Pengembang Layanan Manajemen AWS Utama.

Mengenkripsi data menggunakan kunci KMS yang dikelola pelanggan

Gunakan EncryptionKey API PUTKMS Amazon Fraud Detector untuk mengenkripsi data Amazon Fraud Detector Anda saat istirahat menggunakan kunci KMS yang dikelola pelanggan. Anda dapat mengubah konfigurasi enkripsi kapan saja menggunakan `PutKMSEncryptionKey` API.

Catatan penting tentang data terenkripsi

- Data yang dihasilkan setelah menyiapkan kunci KMS yang dikelola pelanggan dienkripsi. Data yang dihasilkan sebelum menyiapkan kunci KMS yang dikelola pelanggan akan tetap tidak terenkripsi.
- Jika kunci KMS yang dikelola pelanggan diubah, data yang dienkripsi menggunakan konfigurasi enkripsi sebelumnya tidak akan dienkripsi ulang.

Lihat data

Bila Anda menggunakan kunci KMS yang dikelola pelanggan untuk mengenkripsi data Amazon Fraud Detector Anda, data yang dienkripsi menggunakan metode ini tidak dapat dicari menggunakan

filter di area Search Past Predictions di konsol Amazon Fraud Detector. Untuk memastikan hasil pencarian lengkap, gunakan satu atau beberapa properti berikut untuk memfilter hasil:

- ID peristiwa
- Stempel waktu evaluasi
- Status detektor
- Versi detektor
- Versi model
- Jenis model
- Status evaluasi aturan
- Mode eksekusi aturan
- Status pencocokan aturan
- Versi aturan
- Sumber data variabel

Jika kunci KMS yang dikelola pelanggan dihapus atau dijadwalkan untuk dihapus, data Anda mungkin tidak tersedia. Untuk informasi selengkapnya, lihat Menghapus kunci KMS.

Amazon Fraud Detector dan antarmuka VPC endpoint () AWS PrivateLink

Anda dapat membuat koneksi pribadi antara VPC dan Amazon Fraud Detector dengan membuat titik akhir VPC antarmuka. Endpoint antarmuka didukung oleh AWS PrivateLink, teknologi yang memungkinkan Anda mengakses API Amazon Fraud Detector secara pribadi tanpa gateway internet, perangkat NAT, koneksi VPN, atau koneksi AWS Direct Connect. Instans di VPC Anda tidak memerlukan alamat IP publik untuk berkomunikasi dengan Amazon Fraud Detector API. Lalu lintas antara VPC Anda dan Amazon Fraud Detector tidak meninggalkan jaringan Amazon.

Setiap titik akhir antarmuka diwakili oleh satu atau beberapa Antarmuka Jaringan Elastis di subnet Anda.

Untuk informasi selengkapnya, lihat Titik akhir VPC Antarmuka (AWS PrivateLink) di Panduan Pengguna Amazon VPC.

Pertimbangan untuk titik akhir VPC Fraud Detector Amazon

Sebelum menyiapkan titik akhir VPC antarmuka untuk Amazon Fraud Detector, pastikan Anda meninjau properti dan batasan titik akhir Antarmuka di Panduan Pengguna Amazon VPC.

Amazon Fraud Detector mendukung panggilan ke semua tindakan API-nya dari VPC Anda.

Kebijakan titik akhir VPC didukung untuk Amazon Fraud Detector. Secara default, akses penuh ke Amazon Fraud Detector diizinkan melalui titik akhir. Untuk informasi selengkapnya, lihat Mengontrol akses ke layanan dengan titik akhir VPC dalam Panduan Pengguna Amazon VPC.

Membuat titik akhir VPC antarmuka untuk Amazon Fraud Detector

Anda dapat membuat titik akhir VPC untuk layanan Amazon Fraud Detector menggunakan konsol Amazon VPC atau (). AWS Command Line Interface AWS CLI Untuk informasi selengkapnya, lihat Membuat titik akhir antarmuka dalam Panduan Pengguna Amazon VPC.

Buat titik akhir VPC untuk Amazon Fraud Detector menggunakan nama layanan berikut:

- com.amazonaws. *wilayah .frauddetector*

Jika Anda mengaktifkan DNS pribadi untuk titik akhir, Anda dapat membuat permintaan API ke Amazon Fraud Detector menggunakan nama DNS default untuk Wilayah, misalnya,. `frauddetector.us-east-1.amazonaws.com`

Untuk informasi selengkapnya, lihat Mengakses layanan melalui titik akhir antarmuka dalam Panduan Pengguna Amazon VPC.

Membuat kebijakan endpoint VPC untuk Amazon Fraud Detector

Anda dapat membuat kebijakan untuk titik akhir VPC antarmuka untuk Amazon Fraud Detector untuk menentukan hal berikut:

- 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 titik akhir VPC berikut menetapkan bahwa semua pengguna yang memiliki akses ke titik akhir antarmuka VPC diizinkan untuk mengakses detektor Detektor Fraud Amazon bernama. `my_detector`

```
{
```

```
  "Statement": [
      {
          "Action": "frauddetector:*Detector",
          "Effect": "Allow",
          "Resource": "arn:aws:frauddetector:us-east-1:123456789012:detector/
my_detector",
          "Principal": "*"
      }
  ]
}
```

Dalam contoh ini, berikut ini ditolak:

- Tindakan API Fraud Detector Amazon lainnya
- Menggunakan Amazon Fraud Detector API `GetEventPrediction`

> **Note**
>
> Dalam contoh ini, pengguna masih dapat mengambil tindakan Amazon Fraud Detector API lainnya dari luar VPC. Untuk informasi tentang cara membatasi panggilan API ke panggilan dari dalam VPC, lihat. Kebijakan berbasis identitas Amazon Fraud Detector

Memilih untuk tidak menggunakan data Anda untuk perbaikan layanan

Data peristiwa historis yang Anda berikan untuk melatih model dan menghasilkan prediksi digunakan semata-mata untuk menyediakan dan memelihara layanan Anda. Data ini juga dapat digunakan untuk meningkatkan kualitas Amazon Fraud Detector. Kepercayaan, privasi, dan keamanan konten Anda adalah prioritas utama kami dan memastikan bahwa penggunaan kami sesuai dengan komitmen kami kepada Anda. Lihat FAQ Privasi Data untuk informasi selengkapnya

Anda dapat memilih untuk tidak menggunakan data acara Anda untuk mengembangkan atau meningkatkan kualitas Amazon Fraud Detector dengan mengunjungi halaman kebijakan penolakan layanan AI di Panduan Pengguna AWS Organizations dan mengikuti proses yang dijelaskan di sana.

> **Note**
>
> Akun AWS Anda harus dikelola secara terpusat oleh AWS Organizations agar Anda dapat menggunakan kebijakan opt-out. Jika Anda belum membuat organisasi untuk akun AWS

Anda, kunjungi Membuat dan mengelola halaman organisasi dan ikuti proses yang dijelaskan di sana.

Manajemen identitas dan akses untuk Amazon Fraud Detector

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke sumber daya AWS secara aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya Amazon Fraud Detector. IAM adalah layanan Layanan AWS yang dapat Anda gunakan tanpa dikenakan biaya tambahan.

Topik

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan di Amazon Fraud Detector.

Pengguna layanan - Jika Anda menggunakan layanan Amazon Fraud Detector untuk melakukan pekerjaan Anda, administrator Anda akan memberi Anda kredensi dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak fitur Amazon Fraud Detector untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara akses dikelola dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di Amazon Fraud Detector, lihatMemecahkan masalah identitas dan akses Amazon Fraud Detector.

Administrator layanan - Jika Anda bertanggung jawab atas sumber daya Amazon Fraud Detector di perusahaan Anda, Anda mungkin memiliki akses penuh ke Amazon Fraud Detector. Tugas Anda

adalah menentukan fitur dan sumber daya Amazon Fraud Detector mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM Anda untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep Basic IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakan IAM dengan Amazon Fraud Detector, lihatBagaimana Amazon Fraud Detector bekerja dengan IAM.

Administrator IAM - Jika Anda administrator IAM, Anda mungkin ingin mempelajari detail tentang cara menulis kebijakan untuk mengelola akses ke Amazon Fraud Detector. Untuk melihat contoh kebijakan berbasis identitas Amazon Fraud Detector yang dapat Anda gunakan di IAM, lihat. Contoh kebijakan berbasis identitas Amazon Fraud Detector

Mengautentikasi dengan identitas

Autentikasi adalah cara Anda untuk masuk ke AWS menggunakan kredensial identitas Anda. Anda harus terautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai pengguna IAM, atau dengan mengambil peran IAM.

Anda dapat masuk ke AWS sebagai identitas terfederasi dengan menggunakan kredensial yang disediakan melalui sumber identitas. Pengguna AWS IAM Identity Center Pengguna (Pusat Identitas IAM), autentikasi Single Sign-On perusahaan Anda, dan kredensial Google atau Facebook Anda merupakan contoh identitas terfederasi. Saat Anda masuk sebagai identitas gabungan, administrator Anda sebelumnya menyiapkan federasi identitas menggunakan peran IAM. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil suatu peran.

Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal akses AWS. Untuk informasi selengkapnya tentang cara masuk ke AWS, lihat Cara masuk ke Akun AWS dalam Panduan Pengguna AWS Sign-In.

Jika Anda mengakses AWS secara terprogram, AWS memberikan Kit Pengembangan Perangkat Lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensial Anda. Jika Anda tidak menggunakan peralatan AWS, Anda harus menandatangani permintaan sendiri. Untuk informasi selengkapnya tentang cara menggunakan metode yang disarankan untuk menandatangani permintaan sendiri, lihat Menandatangani permintaan API AWS dalam Panduan Pengguna IAM.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Sebagai contoh, AWS menyarankan Anda menggunakan autentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari lebih lanjut, lihat

Autentikasi multi-faktor dalam Panduan Pengguna AWS IAM Identity Center dan Menggunakan autentikasi multi-faktor (MFA) di AWS dalam Panduan Pengguna IAM.

Pengguna root Akun AWS

Ketika membuat Akun AWS, Anda memulai dengan satu identitas masuk yang memiliki akses penuh ke semua Layanan AWS dan sumber daya di akun tersebut. Identitas ini disebut pengguna root Akun AWS dan diakses dengan cara masuk menggunakan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari Anda. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar tugas lengkap yang mengharuskan Anda masuk sebagai pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Pengguna dan grup

Pengguna IAM adalah identitas dalam Akun AWS Anda yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, sebaiknya andalkan kredensial temporer, dan bukan membuat pengguna IAM yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan khusus yang memerlukan kredensial jangka panjang dengan pengguna IAM, sebaiknya rotasikan kunci akses. Untuk informasi selengkapnya, lihat Merotasi kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensial jangka panjang dalam Panduan Pengguna IAM.

Grup IAM adalah identitas yang menentukan kumpulan pengguna IAM. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin untuk beberapa pengguna sekaligus. Grup membuat izin lebih mudah dikelola untuk sekelompok besar pengguna. Misalnya, Anda dapat memiliki grup yang bernama IAMAdmins dan memberikan izin kepada grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran tersebut dimaksudkan untuk dapat diambil oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari selengkapnya, silakan lihat Kapan harus membuat pengguna IAM (bukan peran) dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM merupakan identitas dalam Akun AWS Anda yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Anda dapat mengambil peran IAM

untuk sementara dalam AWS Management Console dengan berganti peran. Anda dapat mengambil peran dengan cara memanggil operasi API AWS CLI atau AWS atau menggunakan URL kustom. Untuk informasi selengkapnya tentang metode untuk menggunakan peran, lihat Menggunakan peran IAM dalam Panduan Pengguna IAM.

Peran IAM dengan kredensial sementara berguna dalam situasi berikut:

- Akses pengguna gabungan – Untuk menetapkan izin ke sebuah identitas gabungan, Anda dapat membuat peran dan menentukan izin untuk peran tersebut. Saat identitas terfederasi diautentikasi, identitas tersebut dikaitkan dengan peran dan diberikan izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat Membuat peran untuk Penyedia Identitas pihak ketiga dalam Panduan Pengguna IAM. Jika Anda menggunakan Pusat Identitas IAM, Anda mengonfigurasi sekumpulan izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM mengaitkan izin yang ditetapkan ke peran dalam IAM. Untuk informasi tentang rangkaian izin, lihat Rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center.
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM guna mendapatkan berbagai izin secara sementara untuk tugas tertentu.
- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (pengguna utama tepercaya) dengan akun berbeda untuk mengakses sumber daya yang ada di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, pada beberapa Layanan AWS, Anda dapat menyertakan kebijakan secara langsung ke sumber daya (bukan menggunakan peran sebagai proksi). Untuk mempelajari perbedaan antara kebijakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Bagaimana peran IAM berbeda dari kebijakan berbasis sumber daya dalam Panduan Pengguna IAM.
- Akses lintas layanan – Sebagian Layanan AWS menggunakan fitur di Layanan AWS lainnya. Contoh, ketika Anda melakukan panggilan dalam layanan, umumnya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Suatu layanan mungkin melakukan hal tersebut menggunakan izin pengguna utama panggilan, menggunakan peran layanan, atau peran terkait layanan.
  - Sesi akses maju (FAS) – Ketika Anda menggunakan pengguna IAM atau peran IAM untuk melakukan tindakan di AWS, Anda akan dianggap sebagai seorang pengguna utama. Saat menggunakan beberapa layanan, Anda mungkin melakukan tindakan yang kemudian dilanjutkan oleh tindakan lain pada layanan yang berbeda. FAS menggunakan izin dari pengguna utama untuk memanggil Layanan AWS, yang dikombinasikan dengan Layanan AWS yang diminta untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya diajukan saat layanan

menerima permintaan yang memerlukan interaksi dengan Layanan AWS lain atau sumber daya lain untuk diselesaikan. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Meneruskan sesi akses.

- Peran IAM – Peran layanan adalah peran IAM yang diambil layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Membuat peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.
- Peran terkait layanan – Peran terkait layanan adalah tipe peran layanan yang terkait dengan Layanan AWS. Layanan tersebut dapat mengambil peran untuk melakukan sebuah tindakan atas nama Anda. Peran terkait layanan akan muncul di Akun AWS Anda dan dimiliki oleh layanan tersebut. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

- Aplikasi yang berjalan di Amazon EC2 – Anda dapat menggunakan peran IAM untuk mengelola kredensial sementara untuk aplikasi yang berjalan di instans EC2 dan mengajukan permintaan API AWS CLI atau AWS. Cara ini lebih dianjurkan daripada menyimpan kunci akses dalam instans EC2. Untuk menetapkan peran AWS ke instans EC2 dan menyediakannya bagi semua aplikasinya, Anda dapat membuat profil instans yang dilampirkan ke instans tersebut. Profil instans berisi peran dan memungkinkan program yang berjalan di instans EC2 mendapatkan kredensial sementara. Untuk informasi selengkapnya, lihat Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di instans Amazon EC2 dalam Panduan Pengguna IAM.

Untuk mempelajari apakah kita harus menggunakan peran IAM atau pengguna IAM, lihat Kapan harus membuat peran IAM (bukan pengguna) dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengendalikan akses di AWS dengan membuat kebijakan dan melampirkannya ke identitas atau sumber daya AWS. Kebijakan adalah objek di AWS yang, ketika terkait dengan identitas atau sumber daya, akan menentukan izinnya. AWS mengevaluasi kebijakan-kebijakan tersebut ketika seorang pengguna utama (pengguna, pengguna root, atau sesi peran) mengajukan permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan di AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang struktur dan isi dokumen kebijakan JSON, silakan lihat Gambaran Umum kebijakan JSON dalam Panduan Pengguna IAM.

Administrator dapat menggunakan kebijakan JSON AWS untuk menentukan secara spesifik siapa yang memiliki akses terhadap apa. Artinya, pengguna utama manakah yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan pada sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat menjalankan peran.

Kebijakan IAM mendefinisikan izin untuk suatu tindakan terlepas dari metode yang Anda gunakan untuk operasi. Sebagai contoh, anggap saja Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut dapat memperoleh informasi peran dari AWS Management Console, AWS CLI, atau API AWS.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan pengguna dan peran, di sumber daya mana, dan dengan ketentuan apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan terkelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran di Akun AWS Anda. Kebijakan terkelola meliputi kebijakan yang dikelola AWS dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan terkelola atau kebijakan inline, lihat Memilih antara kebijakan terkelola dan kebijakan inline dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya yang dilampiri kebijakan tersebut, kebijakan ini menentukan jenis tindakan yang dapat dilakukan oleh pengguna utama tertentu di sumber daya tersebut dan apa ketentuannya. Anda harus menentukan pengguna utama dalam kebijakan berbasis sumber daya. Pengguna utama dapat mencakup akun, pengguna, peran, pengguna gabungan, atau Layanan AWS.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan yang dikelola AWS dari IAM dalam kebijakan berbasis sumber daya.

Daftar Kontrol Akses (ACL)

Daftar kontrol akses (ACL) mengendalikan pengguna utama mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACL serupa dengan kebijakan berbasis sumber daya, meskipun tidak menggunakan format dokumen kebijakan JSON.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung ACL. Untuk mempelajari ACL selengkapnya, silakan lihat Gambaran umum daftar kontrol akses (ACL) di Panduan Developer Layanan Penyimpanan Ringkas Amazon.

Tipe kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Tipe-tipe kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda berdasarkan tipe kebijakan yang lebih umum.

- Batasan izin – Batasan izin adalah fitur lanjutan di mana Anda menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM (pengguna atau peran IAM). Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan secara eksplisit terhadap salah satu kebijakan ini akan mengesampingkan izin tersebut. Untuk informasi selengkapnya tentang batasan izin, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCP) – SCP adalah kebijakan JSON yang menentukan izin maksimum untuk sebuah organisasi atau unit organisasi (OU) di AWS Organizations. AWS Organizations adalah layanan untuk mengelompokkan dan mengelola beberapa akun Akun AWS yang dimiliki bisnis Anda secara terpusat. Jika Anda mengaktifkan semua fitur di organisasi, Anda dapat menerapkan kebijakan kontrol layanan (SCP) ke salah satu atau semua akun Anda. SCP membatasi izin untuk entitas dalam akun anggota, termasuk setiap Pengguna root akun AWS. Untuk informasi selengkapnya tentang Organisasi dan SCP, lihat Cara kerja SCP dalam Panduan Pengguna AWS Organizations.
- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda teruskan sebagai parameter saat Anda membuat sesi sementara secara terprogram untuk peran atau pengguna gabungan. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran

dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan eksplisit di salah satu kebijakan ini akan membatalkan izin tersebut. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Jika beberapa jenis kebijakan diberlakukan untuk satu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan ketika ada beberapa jenis kebijakan, lihat Logika evaluasi kebijakan dalam Panduan Pengguna IAM.

Bagaimana Amazon Fraud Detector bekerja dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses ke Amazon Fraud Detector, Anda harus memahami fitur IAM apa yang tersedia untuk digunakan dengan Amazon Fraud Detector. Untuk mendapatkan tampilan tingkat tinggi tentang cara kerja Amazon Fraud Detector dan AWS layanan lainnya dengan IAM, lihat AWSLayanan yang Bekerja dengan IAM di Panduan Pengguna IAM.

Topik

- Kebijakan berbasis identitas Amazon Fraud Detector
- Kebijakan berbasis sumber daya Amazon Fraud Detector
- Otorisasi Berdasarkan Tag Fraud Detector Amazon
- Peran IAM Detektor Fraud Detector Amazon

Kebijakan berbasis identitas Amazon Fraud Detector

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan tindakan dan sumber daya yang diizinkan atau ditolak, serta ketentuan terkait jenis tindakan yang diizinkan atau ditolak. Amazon Fraud Detector mendukung tindakan, sumber daya, dan kunci kondisi tertentu. Untuk mempelajari semua elemen yang Anda gunakan dalam kebijakan JSON, lihat Referensi Elemen Kebijakan JSON IAM dalam Panduan Pengguna IAM.

Untuk memulai dengan Amazon Fraud Detector, sebaiknya buat pengguna dengan akses terbatas pada operasi Amazon Fraud Detector dan izin yang diperlukan. Anda dapat menambahkan izin lain sesuai kebutuhan. Kebijakan berikut memberikan izin yang diperlukan untuk menggunakan Amazon Fraud Detector: `AmazonFraudDetectorFullAccessPolicy` dan`AmazonS3FullAccess`. Untuk informasi selengkapnya tentang cara menyiapkan Amazon Fraud Detector menggunakan kebijakan ini, lihatSiapkan untuk Amazon Fraud Detector.

Tindakan

Administrator dapat menggunakan kebijakan JSON AWS untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama seperti operasi API AWS terkait. Ada beberapa pengecualian, misalnya tindakan hanya izin yang tidak memiliki operasi API yang cocok. Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Menyertakan tindakan dalam suatu kebijakan untuk memberikan izin melakukan operasi terkait.

Tindakan kebijakan di Amazon Fraud Detector menggunakan awalan berikut sebelum tindakan:`frauddetector:`. Misalnya, untuk membuat aturan dengan operasi Amazon Fraud Detector `CreateRule` API, Anda menyertakan `frauddetector:CreateRule` tindakan tersebut dalam kebijakan. Pernyataan kebijakan harus memuat elemen `Action` atau `NotAction`. Amazon Fraud Detector mendefinisikan serangkaian tindakannya sendiri yang menjelaskan tugas yang dapat Anda lakukan dengan layanan ini.

Untuk menetapkan beberapa tindakan dalam satu pernyataan, pisahkan dengan koma seperti berikut:

```
"Action": [
      "frauddetector:action1",
      "frauddetector:action2"
```

Anda dapat menentukan beberapa tindakan menggunakan wildcard (*). Sebagai contoh, untuk menentukan semua tindakan yang dimulai dengan kata `Describe`, sertakan tindakan berikut:

```
"Action": "frauddetector:Describe*"
```

Untuk melihat daftar tindakan Amazon Fraud Detector, lihat Tindakan yang Ditentukan oleh Amazon Fraud Detector di Panduan Pengguna IAM.

Sumber daya

Administrator dapat menggunakan kebijakan JSON AWS untuk menentukan siapa yang memiliki akses ke apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek atau beberapa objek yang menjadi target penerapan tindakan. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Anda dapat melakukan ini untuk tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (*) untuk mengindikasikan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*"
```

Jenis Sumber Daya yang Ditentukan oleh Amazon Fraud Detector mencantumkan semua ARN sumber daya Amazon Fraud Detector.

Misalnya, untuk menentukan `my_detector` detektor dalam pernyataan Anda, gunakan ARN berikut:

```
"Resource": "arn:aws:frauddetector:us-east-1:123456789012:detector/my_detector"
```

Untuk informasi lebih lanjut tentang format ARN, lihat Amazon Resource Name (ARN) dan Namespace Layanan AWS.

Untuk menentukan semua detektor milik akun tertentu, gunakan wildcard (*):

```
"Resource": "arn:aws:frauddetector:us-east-1:123456789012:detector/*"
```

Beberapa tindakan Amazon Fraud Detector, seperti untuk membuat sumber daya, tidak dapat dilakukan pada sumber daya tertentu. Dalam kasus tersebut, Anda harus menggunakan wildcard (*).

```
"Resource": "*"
```

Untuk melihat daftar jenis sumber daya Amazon Fraud Detector dan ARNnya, lihat Sumber Daya yang Ditentukan oleh Amazon Fraud Detector di Panduan Pengguna IAM. Untuk mempelajari

tindakan mana yang dapat Anda tentukan ARN dari setiap sumber daya, lihat Tindakan yang Ditentukan oleh Amazon Fraud Detector.

Kunci syarat

Administrator dapat menggunakan kebijakan JSON AWS untuk menentukan siapa yang memiliki akses ke hal apa. Yaitu, pengguna utama mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Elemen `Condition` (atau blok `Condition`) memungkinkan Anda menentukan kondisi di mana suatu pernyataan akan diterapkan. Elemen `Condition` bersifat opsional. Anda dapat membuat ekspresi kondisional yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta.

Jika Anda menentukan beberapa elemen `Condition` dalam satu pernyataan, atau beberapa kunci dalam satu elemen `Condition`, AWS akan mengevaluasinya dengan menggunakan operasi `AND` logis. Jika Anda menentukan beberapa niilai untuk satu kunci persyaratan, AWS akan mengevaluasi syarat tersebut menggunakan operasi `OR` yang logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Sebagai contoh, Anda dapat memberikan izin kepada pengguna IAM untuk mengakses sumber daya hanya jika izin tersebut mempunyai tanda yang sesuai dengan nama pengguna IAM mereka. Untuk informasi selengkapnya, silakan lihat Elemen kebijakan IAM: variabel dan tanda di Panduan Pengguna IAM.

AWS mendukung kunci kondisi global dan kunci kondisi spesifik layanan. Untuk melihat semua kunci kondisi global AWS, lihat kunci konteks kondisi global AWS dalam Panduan Pengguna IAM.

Amazon Fraud Detector mendefinisikan set kunci kondisinya sendiri dan juga mendukung penggunaan beberapa kunci kondisi global. Untuk melihat semua kunci syarat global AWS, lihat Kunci Konteks Syarat Global AWS dalam Panduan Pengguna IAM.

Untuk melihat daftar kunci kondisi Amazon Fraud Detector, lihat Kunci Kondisi untuk Amazon Fraud Detector di Panduan Pengguna IAM. Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat Tindakan yang Ditentukan oleh Amazon Fraud Detector.

Contoh-contoh

Untuk melihat contoh kebijakan berbasis identitas Amazon Fraud Detector, lihat. Contoh kebijakan berbasis identitas Amazon Fraud Detector

Kebijakan berbasis sumber daya Amazon Fraud Detector

Amazon Fraud Detector tidak mendukung kebijakan berbasis sumber daya.

Otorisasi Berdasarkan Tag Fraud Detector Amazon

Anda dapat melampirkan tag ke sumber daya Amazon Fraud Detector atau meneruskan tag dalam permintaan ke Amazon Fraud Detector. Untuk mengendalikan akses berdasarkan tag, berikan informasi tentang tanda di elemen syarat dari sebuh kebijakan dengan menggunakan kunci-kunci persyaratan `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`.

Peran IAM Detektor Fraud Detector Amazon

IAM role adalah entitas di dalam akun AWS Anda yang memiliki izin tertentu.

Menggunakan kredensyal sementara dengan Amazon Fraud Detector

Anda dapat menggunakan kredensial sementara untuk masuk dengan gabungan, menjalankan IAM role, atau menjalankan peran lintas akun. Anda memperoleh kredensil keamanan sementara dengan memanggil operasi AWS STS API seperti AssumeRoleatau. GetFederationToken

Amazon Fraud Detector mendukung penggunaan kredensyal sementara.

Peran terkait layanan

Peran terkait layanan mengizinkan layanan AWS untuk mengakses sumber daya di layanan lain untuk menyelesaikan tindakan atas nama Anda. Peran terkait layanan muncul di akun IAM Anda dan dimiliki oleh layanan tersebut. Administrator IAM dapat melihat tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Amazon Fraud Detector tidak mendukung peran terkait layanan.

Peran layanan

Fitur ini memungkinkan layanan untuk menerima peran layanan atas nama Anda. Peran ini mengizinkan layanan mengakses sumber daya di layanan lain untuk menyelesaikan tindakan atas nama Anda. Peran layanan muncul di akun Anda dan dimiliki oleh akun tersebut. Ini berarti bahwa administrator dapat mengubah izin untuk peran ini. Namun, melakukannya mungkin merusak fungsi layanan.

Amazon Fraud Detector mendukung peran layanan.

Contoh kebijakan berbasis identitas Amazon Fraud Detector

Secara default, pengguna dan peran IAM tidak memiliki izin untuk membuat atau memodifikasi sumber daya Amazon Fraud Detector. Mereka juga tidak dapat melakukan tugas dengan menggunakan AWS Management Console, AWS CLI, atau API AWS. Administrator harus membuat kebijakan IAM yang memberikan izin kepada pengguna dan peran untuk melakukan operasi API tertentu pada sumber daya tertentu yang mereka butuhkan. Administrator kemudian harus melampirkan kebijakan tersebut ke pengguna atau grup yang memerlukan izin tersebut.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat Kebijakan pada Tab JSON dalam Panduan Pengguna IAM.

Topik

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya Amazon Fraud Detector di akun Anda. Tindakan ini dikenai biaya untuk Akun AWS Anda. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulai menggunakan kebijakan yang dikelola AWS dan beralih ke izin dengan hak akses paling rendah – Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan yang dikelola AWS yang memberikan izin untuk banyak kasus penggunaan umum. Kebijakan ini ada di Akun AWS Anda. Sebaiknya Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola pelanggan AWS yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat kebijakan yang dikelola AWS atau kebijakan yang dikelola AWS untuk fungsi pekerjaan di Panduan Pengguna IAM.

- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukan ini dengan menentukan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, juga dikenal sebagai izin hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk menerapkan izin, lihat Kebijakan dan izin di IAM di Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Misalnya, Anda dapat menulis syarat kebijakan untuk menentukan bahwa semua pengajuan harus dikirim menggunakan SSL. Anda juga dapat menggunakan kondisi untuk memberi akses ke tindakan layanan jika digunakan melalui Layanan AWS yang spesifk, seperti AWS CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Syarat di Panduan Pengguna IAM.
- Menggunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda guna memastikan izin yang aman dan berfungsi – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat validasi kebijakan Analizer Akses IAM di Panduan Pengguna IAM.
- Wajibkan autentikasi multi-faktor (MFA) – Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Akun AWS Anda, aktifkan MFA untuk keamanan tambahan. Untuk mewajibkan MFA saat operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Mengonfigurasi akses API yang dilindungi MFA di Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat Praktik terbaik keamanan di IAM di Panduan Pengguna IAM.

Kebijakan AWS yang dikelola (telah ditentukan sebelumnya) untuk Amazon Fraud Detector

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 tidak perlu menyelidiki izin mana yang diperlukan. Untuk informasi selengkapnya, lihat Kebijakan Terkelola AWS di Panduan Pengguna AWS Identity and Access Management Manajemen.

Kebijakan AWS terkelola berikut, yang dapat Anda lampirkan ke pengguna di akun Anda, khusus untuk Amazon Fraud Detector:

`AmazonFraudDetectorFullAccess`: Memberikan akses penuh ke sumber daya Amazon Fraud Detector, tindakan, dan operasi yang didukung termasuk:

- Buat daftar dan jelaskan semua titik akhir model di Amazon SageMaker
- Buat daftar semua peran IAM di akun
- Daftar semua ember Amazon S3
- Izinkan IAM Pass Role untuk meneruskan peran ke Amazon Fraud Detector

Kebijakan ini tidak menyediakan akses S3 yang tidak dibatasi. Jika Anda perlu mengunggah kumpulan data pelatihan model ke S3, kebijakan `AmazonS3FullAccess` terkelola (atau kebijakan akses Amazon S3 kustom cakupan) juga diperlukan.

Anda dapat meninjau izin kebijakan dengan masuk ke konsol IAM dan mencari berdasarkan nama kebijakan. Anda juga dapat membuat kebijakan IAM kustom Anda sendiri untuk mengizinkan izin untuk tindakan dan sumber daya Amazon Fraud Detector saat Anda membutuhkannya. Anda dapat melampirkan kebijakan khusus ini ke pengguna atau grup yang memerlukannya.

Izinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan para pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan pada konsol atau menggunakan AWS CLI atau AWS API secara terprogram.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
```

```
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Izinkan akses penuh ke sumber daya Amazon Fraud Detector

Contoh berikut memberi pengguna akses Akun AWS penuh ke semua sumber daya dan tindakan Amazon Fraud Detector.

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

Izinkan akses hanya-baca ke sumber daya Amazon Fraud Detector

Dalam contoh ini, Anda memberi pengguna akses Akun AWS hanya-baca ke sumber daya Amazon Fraud Detector Anda.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "frauddetector:GetEventTypes",
                "frauddetector:BatchGetVariable",
                "frauddetector:DescribeDetector",
                "frauddetector:GetModelVersion",
                "frauddetector:GetEventPrediction",
                "frauddetector:GetExternalModels",
                "frauddetector:GetLabels",
                "frauddetector:GetVariables",
                "frauddetector:GetDetectors",
                "frauddetector:GetRules",
                "frauddetector:ListTagsForResource",
                "frauddetector:GetKMSEncryptionKey",
                "frauddetector:DescribeModelVersions",
                "frauddetector:GetDetectorVersion",
                "frauddetector:GetPrediction",
                "frauddetector:GetOutcomes",
                "frauddetector:GetEntityTypes",
                "frauddetector:GetModels"
            ],
            "Resource": "*"
        }
    ]
}
```

Izinkan akses ke sumber daya tertentu

Dalam contoh kebijakan tingkat sumber daya ini, Anda memberi pengguna Akun AWS akses ke semua tindakan dan sumber daya kecuali untuk satu sumber daya Detektor tertentu.

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

```
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "frauddetector:*Detector"
            ],
            "Resource": "arn:${Partition}:frauddetector:${Region}:${Account}:detector/
${detector-name}"
        }
    ]
}
```

Izinkan akses ke sumber daya tertentu saat menggunakan API mode ganda

Amazon Fraud Detector menyediakan dual mode get API yang berfungsi sebagai operasi List dan Description. API mode ganda saat dipanggil tanpa parameter apa pun mengembalikan daftar sumber daya tertentu yang terkait dengan AndaAkun AWS. API mode ganda saat dipanggil dengan parameter mengembalikan detail sumber daya yang ditentukan. Sumber daya dapat berupa model, variabel, jenis peristiwa, atau tipe entitas.

API mode ganda mendukung izin tingkat sumber daya dalam kebijakan IAM. Namun, izin tingkat sumber daya hanya diterapkan ketika satu atau beberapa parameter disediakan sebagai bagian dari permintaan. Misalnya, jika pengguna memanggil GetVariablesAPI dan memberikan nama variabel dan jika ada kebijakan IAM Deny yang dilampirkan ke sumber daya variabel atau nama variabel, pengguna akan menerima `AccessDeniedException` kesalahan. Jika pengguna memanggil `GetVariables` API dan tidak menentukan nama variabel, semua variabel dikembalikan, yang dapat menyebabkan kebocoran informasi.

Untuk memungkinkan pengguna melihat detail sumber daya tertentu saja, gunakan elemen `NotResource` kebijakan IAM dalam kebijakan IAM Deny. Setelah Anda menambahkan elemen kebijakan ini ke kebijakan IAM Deny, pengguna hanya dapat melihat detail sumber daya yang ditentukan dalam `NotResource` blok. Untuk informasi selengkapnya, lihat elemen kebijakan IAM JSON: NotResource di Panduan Pengguna IAM.

Contoh kebijakan berikut memungkinkan pengguna untuk mengakses semua sumber daya Amazon Fraud Detector. Namun, elemen `NotResource` kebijakan digunakan untuk membatasi panggilan GetVariablesAPI hanya pada nama variabel dengan awalan `user*job_*`, dan. `var*`

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

```
 "Statement": [
  {
    "Effect": "Allow",
    "Action": "frauddetector:*",
    "Resource": "*"
  },
 {
    "Effect": "Deny",
    "Action": "frauddetector:GetVariables",
    "NotResource": [
       "arn:aws:frauddetector:*:*:variable/user*",
       "arn:aws:frauddetector:*:*:variable/job_*",
       "arn:aws:frauddetector:*:*:variable/var*"
    ]
  }
 ]
}
```

Respons

Untuk kebijakan contoh ini, respons menunjukkan perilaku berikut:

- GetVariables Panggilan yang tidak menyertakan nama variabel menghasilkan `AccessDeniedException` kesalahan karena permintaan dipetakan ke pernyataan Deny.
- GetVariables Panggilan yang menyertakan nama variabel yang tidak diizinkan, menghasilkan `AccessDeniedException` kesalahan karena nama variabel tidak dipetakan ke nama variabel di `NotResource` blok. Misalnya, GetVariables panggilan dengan nama variabel `email_address` menghasilkan `AccessDeniedException` kesalahan.
- GetVariables Panggilan yang menyertakan nama variabel yang cocok dengan nama variabel di `NotResource` blok dikembalikan seperti yang diharapkan. Misalnya, GetVariables panggilan yang menyertakan nama variabel `job_cpa` mengembalikan rincian `job_cpa` variabel.

Membatasi akses berdasarkan tag

Kebijakan contoh ini menunjukkan cara membatasi akses ke Amazon Fraud Detector berdasarkan tag sumber daya. Contoh ini mengasumsikan bahwa:

- Dalam Anda, Akun AWS Anda telah mendefinisikan dua grup yang berbeda, bernama Team1 dan Team2

- Anda telah membuat empat detektor
- Anda ingin mengizinkan anggota Team1 melakukan panggilan API pada 2 detektor
- Anda ingin mengizinkan anggota Team2 melakukan panggilan API pada 2 detektor lainnya

Untuk mengontrol akses ke panggilan API (contoh)

1. Tambahkan tag dengan kunci `Project` dan nilai A ke detektor yang digunakan oleh Team1.
2. Tambahkan tag dengan kunci `Project` dan nilai B ke detektor yang digunakan oleh Team2.
3. Buat kebijakan IAM dengan `ResourceTag` kondisi yang menolak akses ke detektor yang memiliki tag dengan kunci `Project` dan nilaiB, dan lampirkan kebijakan tersebut ke Team1.
4. Buat kebijakan IAM dengan `ResourceTag` kondisi yang menolak akses ke detektor yang memiliki tag dengan kunci `Project` dan nilaiA, dan lampirkan kebijakan tersebut ke Team2.

Berikut ini adalah contoh kebijakan yang menolak tindakan spesifik pada sumber daya Amazon Fraud Detector yang memiliki tag dengan kunci `Project` dan nilaiB:

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

      "Action": [

        "frauddetector:CreateModel",
        "frauddetector:CancelBatchPredictionJob",
        "frauddetector:CreateBatchPredictionJob",
        "frauddetector:DeleteBatchPredictionJob",
        "frauddetector:DeleteDetector"
      ],

      "Resource": "*",
      "Condition": {
        "StringEquals": {
```

```
          "aws:ResourceTag/Project": "B"
        }
      }
    }
  ]
}
```

Pencegahan Deputi Bingung

Masalah deputi yang membingungkan terjadi ketika entitas yang tidak memiliki izin untuk melakukan tindakan dapat memaksa entitas yang lebih istimewa untuk melakukan tindakan. AWSmenyediakan alat yang membantu Anda melindungi akun Anda jika Anda memberikan pihak ketiga (disebut lintas akun) atau AWS layanan lain (disebut lintas layanan) akses ke sumber daya di akun Anda.

Masalah wakil kebingungan lintas layanan dapat terjadi ketika satu layanan (layanan panggilan) memanggil layanan lain (layanan yang disebut). Layanan pemanggil dapat dimanipulasi menggunakan izinnya untuk bertindak pada sumber daya pelanggan lain dengan cara yang seharusnya tidak dilakukannya kecuali bila memiliki izin untuk mengakses. Untuk mencegah hal ini, Anda dapat membuat kebijakan yang membantu Anda melindungi data Anda untuk semua layanan dengan prinsipal layanan yang telah diberikan akses ke sumber daya layanan Anda.

Amazon Fraud Detector mendukung penggunaan peran layanan dalam kebijakan izin Anda untuk mengizinkan layanan mengakses sumber daya layanan lain atas nama Anda. Peran memerlukan dua kebijakan: kebijakan kepercayaan peran yang menentukan prinsipal yang diizinkan untuk mengambil peran dan kebijakan izin yang menentukan apa yang dapat dilakukan dengan peran tersebut. Ketika layanan mengambil peran atas nama Anda, kepala layanan harus diizinkan untuk melakukan `sts:AssumeRole` tindakan dalam kebijakan kepercayaan peran. Saat layanan memanggil`sts:AssumeRole`, AWS STS mengembalikan sekumpulan kredensil keamanan sementara yang digunakan oleh prinsipal layanan untuk mengakses sumber daya yang diizinkan oleh kebijakan izin peran.

Untuk mencegah masalah deputi yang membingungkan lintas layanan, Amazon Fraud Detector merekomendasikan penggunaan kunci konteks kondisi `aws:SourceAccount`global `aws:SourceArn`dan global dalam kebijakan kepercayaan peran Anda untuk membatasi akses ke peran hanya pada permintaan yang dihasilkan oleh sumber daya yang diharapkan.

`aws:SourceAccount`Menentukan ID Akun dan `aws:SourceArn` menentukan ARN dari sumber daya yang terkait dengan akses lintas layanan. `aws:SourceArn`Harus ditentukan menggunakan

format ARN. Pastikan `aws:SourceArn` keduanya `aws:SourceAccount` dan menggunakan ID Akun yang sama saat digunakan dalam pernyataan kebijakan yang sama.

Cara paling efektif untuk melindungi dari masalah confused deputy adalah dengan menggunakan kunci konteks kondisi global `aws:SourceArn` dengan ARN sumber daya penuh. Jika Anda tidak mengetahui ARN lengkap sumber daya atau jika Anda menentukan beberapa sumber daya, gunakan kunci kondisi konteks `aws:SourceArn` global dengan wildcard (*) untuk bagian ARN yang tidak diketahui. Misalnya, `arn:aws:servicename:*:123456789012:*`. Untuk informasi tentang sumber daya dan tindakan Amazon Fraud Detector yang dapat Anda gunakan dalam kebijakan izin, lihat Tindakan, sumber daya, dan kunci kondisi untuk Amazon Fraud Detector.

Contoh kebijakan kepercayaan peran berikut menggunakan wildcard (*) di kunci `aws:SourceArn` kondisi untuk memungkinkan Amazon Fraud Detector mengakses beberapa sumber daya yang terkait dengan Id Akun.

```
{
        "Version": "2012-10-17",
        "Statement": [
             {
               "Effect": "Allow",
               "Principal": {
               "Service": [
                   "frauddetector.amazonaws.com"
                   ]
               },
               "Action": "sts:AssumeRole",
               "Condition": {
               "StringEquals": {
                   "aws:SourceAccount": "123456789012"
               },
                 "StringLike": {
                    "aws:SourceArn": "arn:aws:frauddetector:us-west-2:123456789012:*"
        }
      }
    }
  ]
}
```

Kebijakan kepercayaan peran berikut memungkinkan Amazon Fraud Detector mengakses hanya `external-model` sumber daya. Perhatikan `aws:SourceArn` param di blok Kondisi. Kualifikasi

sumber daya dibuat menggunakan titik akhir model yang disediakan untuk melakukan panggilan `PutExternalModel` API.

```
    {
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "frauddetector.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "StringLike": {
          "aws:SourceArn": "arn:aws:frauddetector:us-west-2:123456789012:external-
model/MyExternalModeldoNotDelete-ReadOnly"
        }
      }
    }
  ]
}
```

Memecahkan masalah identitas dan akses Amazon Fraud Detector

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan Amazon Fraud Detector dan IAM.

Topik

- Saya tidak berwenang untuk melakukan tindakan di Amazon Fraud Detector
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar AWS akun saya untuk mengakses sumber daya Amazon Fraud Detector saya
- Amazon Fraud Detector tidak dapat mengambil peran yang diberikan

Saya tidak berwenang untuk melakukan tindakan di Amazon Fraud Detector

Jika AWS Management Console memberi tahu bahwa Anda tidak diberi otorisasi untuk melakukan tindakan, Anda harus menghubungi administrator untuk mendapatkan bantuan. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Contoh kesalahan berikut terjadi ketika `mateojackson` pengguna mencoba menggunakan konsol untuk melihat detail tentang *`detektor`* tetapi tidak memiliki `frauddetector:`*`GetDetectors`* izin.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 frauddetector:GetDetectors on resource: my-example-detector
```

Dalam hal ini, Mateo meminta administratornya untuk memperbarui kebijakannya untuk mengizinkan dia mengakses sumber daya *`my-example-detector`* menggunakan tindakan `frauddetector:`*`GetDetectors`*.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran ke Amazon Fraud Detector.

Sebagian Layanan AWS mengizinkan Anda untuk memberikan peran yang sudah ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait-layanan. Untuk melakukan tindakan tersebut, Anda harus memiliki izin untuk memberikan peran pada layanan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di Amazon Fraud Detector. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda membutuhkan bantuan, hubungi administrator AWS Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang di luar AWS akun saya untuk mengakses sumber daya Amazon Fraud Detector saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau pengguna di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACL), Anda dapat menggunakan kebijakan tersebut untuk memberi pengguna akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa hal berikut:

- Untuk mengetahui apakah Amazon Fraud Detector mendukung fitur-fitur ini, lihatBagaimana Amazon Fraud Detector bekerja dengan IAM.
- Untuk mempelajari cara memberikan akses ke sumber daya di seluruh Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM di Akun AWS lainnya yang Anda miliki dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses ke sumber daya Anda ke pihak ketiga Akun AWS, lihat Menyediakan akses ke akun Akun AWS yang dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (gabungan identitas) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara penggunaan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Perbedaan antara peran IAM dan kebijakan berbasis sumber daya di Panduan Pengguna IAM.

Amazon Fraud Detector tidak dapat mengambil peran yang diberikan

Jika Anda menerima kesalahan bahwa Amazon Fraud Detector tidak dapat mengambil peran yang diberikan, maka Anda harus memperbarui hubungan kepercayaan untuk peran yang ditentukan. Dengan menetapkan Amazon Fraud Detector sebagai entitas tepercaya, layanan dapat mengambil peran tersebut. Saat Anda menggunakan Amazon Fraud Detector untuk membuat peran, hubungan kepercayaan ini diatur secara otomatis. Anda hanya perlu membangun hubungan kepercayaan ini untuk peran IAM yang tidak dibuat oleh Amazon Fraud Detector.

Untuk membangun hubungan kepercayaan untuk peran yang ada dengan Amazon Fraud Detector

1. Buka konsol IAM di https://console.aws.amazon.com/iam/

2. Di panel navigasi pilih Peran.
3. Pilih nama peran yang ingin Anda ubah, dan pilih tab Trust relationship.
4. Pilih Edit hubungan kepercayaan.
5. Di bawah Dokumen Kebijakan, tempel berikut, dan kemudian pilih Perbarui Kebijakan Kepercayaan.

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

Pencatatan dan pemantauan di Amazon Fraud Detector

AWS menyediakan alat pemantauan berikut untuk menonton Amazon Fraud Detector, melaporkan bila ada sesuatu yang salah, dan mengambil tindakan otomatis bila perlu:

- Amazon CloudWatch memantau AWS sumber daya Anda dan aplikasi yang Anda jalankan AWS secara real time. Untuk informasi selengkapnya CloudWatch, lihat Panduan CloudWatch Pengguna Amazon.
- AWS CloudTrail merekam panggilan API dan peristiwa terkait yang dilakukan oleh atau atas nama akun AWS Anda dan mengirimkan berkas log ke bucket Amazon S3 yang Anda tentukan. Untuk informasi selengkapnya CloudTrail, lihat Panduan AWS CloudTrail Pengguna.

Untuk informasi selengkapnya tentang pemantauan Amazon Fraud Detector, lihatPantau Amazon Fraud Detector.

Validasi kepatuhan untuk Amazon Fraud Detector

Auditor pihak ketiga menilai keamanan dan kepatuhan layanan AWS sebagai bagian dari beberapa program kepatuhan AWS, seperti SOC, PCI, FedRAMP, dan HIPAA.

Untuk mempelajari apakah Layanan AWS berada dalam lingkup program kepatuhan khusus, lihat Layanan AWS di Scope oleh Program Program Kepatuhan yang Anda minati. Untuk informasi umum, lihat Program Kepatuhan AWS .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh Laporan di AWS Artifact.

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan berdasarkan sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, serta hukum dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- Panduan Mulai Cepat Keamanan dan Kepatuhan – Panduan deployment ini membahas pertimbangan arsitektur dan menyediakan langkah-langkah untuk melakukan deployment lingkungan dasar di AWS yang menjadi fokus keamanan dan kepatuhan.
- Merancang Keamanan dan Kepatuhan HIPAA di Amazon Web Services – Laporan resmi ini menjelaskan cara perusahaan dapat menggunakan AWS untuk membuat aplikasi yang memenuhi syarat HIPAA.

  > **Note**
  >
  > Tidak semua Layanan AWS memenuhi syarat HIPAA. Untuk informasi selengkapnya, lihat Referensi Layanan yang Memenuhi Syarat HIPAA.

- Sumber Daya Kepatuhan AWS – Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- Panduan Kepatuhan Pelanggan AWS – Pahami model tanggung jawab bersama melalui lensa kepatuhan. Panduan ini merangkum praktik terbaik untuk mengamankan Layanan AWS dan memetakan panduan kontrol keamanan di banyak kerangka kerja (termasuk National Institute of Standards and Technology (NIST), Payment Card Industry Security Standards Council (PCI), dan International Organization for Standardization (ISO)).
- Mengevaluasi Sumber Daya dengan Aturan dalam Panduan Developer AWS Config – Layanan AWS Config menilai seberapa baik konfigurasi sumber daya Anda dalam mematuhi praktik-praktik internal, pedoman industri, dan regulasi internal.

- AWS Security Hub – Layanan AWS ini memberikan pandangan komprehensif tentang status keamanan Anda di dalam AWS. Security Hub menggunakan kontrol keamanan untuk mengevaluasi sumber daya AWS Anda dan memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik. Untuk daftar layanan dan kontrol yang didukung, lihat Referensi kontrol Security Hub.
- AWS Audit Manager – Layanan AWS ini akan membantu Anda untuk terus-menerus mengaudit penggunaan AWS untuk menyederhanakan cara Anda mengelola risiko dan kepatuhan terhadap regulasi dan standar industri.

Ketahanan di Amazon Fraud Detector

Infrastruktur global AWS dibangun di sekitar Wilayah AWS dan Availability Zone. Wilayah AWS menyediakan beberapa Availability Zone yang terpisah dan terisolasi secara fisik, yang terhubung dengan jaringan berlatensi rendah, throughput yang tinggi, dan sangat redundan. Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis mengalami kegagalan di antara zona tanpa gangguan. Zona Ketersediaan lebih tersedia, toleran terhadap kegagalan, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau ganda tradisional.

Untuk informasi lebih lanjut tentang Wilayah AWS dan Zona Ketersediaan, lihat Infrastruktur Global AWS.

Keamanan Infrastruktur di Amazon Fraud Detector

Sebagai layanan terkelola, Amazon Fraud Detector dilindungi oleh keamanan jaringan AWS global. Lihat informasi tentang layanan keamanan AWS dan cara AWS melindungi infrastruktur di Keamanan Cloud AWS. Untuk mendesain lingkungan AWS Anda dengan menggunakan praktik terbaik bagi keamanan infrastruktur, lihat Perlindungan Infrastruktur dalam Pilar Keamanan Kerangka Kerja Berarsitektur Baik AWS.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses Amazon Fraud Detector melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani dengan menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan pengguna utama IAM. Atau Anda dapat menggunakan AWS Security Token Service (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

Pantau Amazon Fraud Detector

Pemantauan adalah bagian penting dalam menjaga keandalan, ketersediaan, dan kinerja Amazon Fraud Detector dan solusi AWS Anda yang lain. AWS menyediakan alat pemantauan berikut untuk menonton Amazon Fraud Detector, melaporkan bila ada sesuatu yang salah, dan mengambil tindakan otomatis bila perlu:

- Amazon CloudWatch memantau AWS sumber daya Anda dan aplikasi yang Anda jalankan AWS secara real time. Anda dapat mengumpulkan dan melacak metrik, membuat dasbor yang disesuaikan, dan mengatur alarm yang memberi tahu Anda atau mengambil tindakan saat metrik tertentu mencapai ambang batas yang ditentukan. Untuk informasi selengkapnya, lihat Panduan CloudWatch Pengguna Amazon.
- AWS CloudTrail merekam panggilan API dan peristiwa terkait yang dilakukan oleh atau atas nama akun AWS Anda dan mengirimkan berkas log ke bucket Amazon S3 yang Anda tentukan. Anda dapat mengidentifikasi pengguna dan akun mana yang memanggil AWS, alamat IP sumber yang melakukan panggilan, dan kapan panggilan tersebut terjadi. Untuk mengetahui informasi selengkapnya, lihat Panduan Pengguna AWS CloudTrail.

Topik

Memantau Amazon Fraud Detector dengan Amazon CloudWatch

Anda dapat memantau Amazon Fraud Detector menggunakan CloudWatch, yang mengumpulkan data mentah dan memprosesnya menjadi metrik hampir real-time yang dapat dibaca. Statistik ini disimpan untuk jangka waktu 15 bulan, sehingga Anda dapat mengakses informasi historis dan mendapatkan perspektif yang lebih baik tentang performa aplikasi atau layanan web Anda. Anda juga dapat mengatur alarm yang memperhatikan ambang batas tertentu dan mengirim notifikasi atau mengambil tindakan saat ambang batas tersebut terpenuhi. Untuk informasi selengkapnya, lihat Panduan CloudWatch Pengguna Amazon.

Topik

Menggunakan CloudWatch Metrik untuk Amazon Fraud Detector.

Untuk menggunakan metrik, Anda harus menentukan informasi berikut:

- Namespace metrik. Namespace adalah wadah yang digunakan CloudWatch Amazon Fraud Detector untuk mempublikasikan metriknya. Jika Anda menggunakan CloudWatch ListMetricsAPI atau perintah list-metrics untuk melihat metrik Amazon Fraud Detector, `AWS/FraudDetector` tentukan namespace.
- Dimensi metrik. Dimensi adalah pasangan nama-nilai yang membantu Anda mengidentifikasi metrik secara unik, misalnya, `DetectorId` dapat berupa nama dimensi. Menentukan dimensi metrik adalah opsional.
- Nama metrik, seperti `GetEventPrediction`.

Anda bisa mendapatkan data pemantauan untuk Amazon Fraud Detector dengan menggunakanAWS Management Console, theAWS CLI, atau CloudWatch API. Anda juga dapat menggunakan CloudWatch API melalui salah satu Kit Pengembangan Perangkat Lunak Amazon AWS (SDK) atau alat CloudWatch API. Konsol menampilkan serangkaian grafik berdasarkan data mentah dari CloudWatch API. Tergantung kebutuhan, Anda mungkin lebih memilih menggunakan grafik yang ditampilkan di konsol atau diterima dari API.

Daftar berikut menunjukkan beberapa penggunaan umum untuk metrik. Berikut ini adalah saran untuk memulai, bukan daftar komprehensif.

Bagaimana caranya?	Metrik Terkait
Bagaimana cara melacak jumlah prediksi yang telah dilakukan?	Pantau `GetEventPrediction` metrik.
Bagaimana saya bisa memantau `GetEventPrediction` kesalahan?	Gunakan `GetEventPrediction5xxError` dan `GetEventPrediction4xxError` metrik.
Bagaimana saya bisa memantau latensi `GetEventPrediction` panggilan?	Gunakan metrik `GetEventPrediction Latency` .

Anda harus memiliki CloudWatch izin yang sesuai untuk memantau Amazon Fraud Detector dengan CloudWatch. Untuk informasi lebih lanjut, lihat Kontrol Autentikasi dan Akses untuk Amazon CloudWatch.

Akses Metrik Detektor Fraud Amazon

Langkah-langkah berikut menunjukkan cara mengakses metrik Amazon Fraud Detector menggunakan CloudWatch konsol.

Cara melihat metrik (konsol)

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Pilih Metrik, pilih tab Semua Metrik, lalu pilih Fraud Detector.
3. Pilih dimensi metrik.
4. Pilih metrik yang diinginkan dari daftar, dan pilih periode waktu untuk grafik.

Buat Alarm

Anda dapat membuat CloudWatch alarm yang mengirimkan pesan Amazon Simple Notification Service (Amazon SNS) saat alarm berubah status. Alarm mengawasi metrik tunggal selama periode waktu yang Anda tentukan. Alarm tersebut melakukan satu atau beberapa tindakan berdasarkan nilai metrik yang relatif terhadap ambang batas tertentu selama beberapa periode waktu. Tindakan ini adalah notifikasi yang dikirim ke topik Amazon SNS atau kebijakan Penskalaan Otomatis.

Alarm memanggil tindakan untuk perubahan status berkelanjutan saja. CloudWatch alarm tidak memanggil tindakan hanya karena mereka berada dalam keadaan tertentu. Status harus diubah dan dipertahankan selama jangka waktu tertentu.

Untuk mengatur alarm (konsol)

1. Masuk ke AWS Management Console dan buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, pilih Alarm, dan pilih Buat Alarm. Ini membuka Create Alarm Wizard.
3. Pilih Pilih Metrik.
4. Di tab Semua metrik, pilih Fraud Detector.
5. Pilih Berdasarkan ID Detektor, lalu pilih GetEventPredictionmetrik.
6. Pilih tab Metrik bergrafik.

7. Untuk Statistik pilih Jumlah.
8. Pilih Pilih Metrik.
9. Untuk Kondisi, pilih Static for Threshold type dan Greater for Wever..., lalu masukkan nilai maksimum pilihan Anda. Pilih Berikutnya.
10. Untuk mengirim alarm ke topik Amazon SNS yang sudah ada, untuk Kirim notifikasi ke:, pilih topik SNS yang sudah ada. Untuk menetapkan nama dan alamat email untuk daftar langganan email baru, pilih Daftar baru. CloudWatch menyimpan daftar dan menampilkannya di bidang sehingga Anda dapat menggunakannya untuk mengatur alarm future.

> **Note**
>
> Jika Anda menggunakan Daftar baru untuk membuat topik Amazon SNS baru, alamat email harus diverifikasi sebelum penerima yang dituju menerima pemberitahuan. Amazon SNS hanya mengirim email ketika alarm memasuki status alarm. Jika perubahan status alarm ini terjadi sebelum alamat email diverifikasi, penerima yang dituju tidak akan menerima pemberitahuan.

11. Pilih Berikutnya. Tambahkan nama dan deskripsi opsional untuk alarm Anda. Pilih Berikutnya.
12. Pilih Buat Alarm.

Metrik Detektor Fraud Amazon

Amazon Fraud Detector mengirimkan metrik berikut ke CloudWatch. Semua metrik mendukung statistik ini:`Average`,, `Minimum``Maximum`,`Sum`.

Metrik	Deskripsi
`GetEventPrediction`	Jumlah permintaan GetEventPrediction API. Dimensi yang Benar: `DetectorID`
`GetEventPredictionLatency`	Interval waktu yang dibutuhkan untuk menanggapi permintaan klien dari GetEventPrediction permintaan. Dimensi yang Benar: `DetectorID`

Metrik	Deskripsi
	Satuan: Milidetik
`GetEventPrediction4XXError`	Jumlah GetEventPrediction permintaan di mana Amazon Fraud Detector mengembalikan kode respons HTTP 4xx. Untuk setiap respons 4xx, 1 dikirim. Dimensi yang Benar: `DetectorID`
`GetEventPrediction5XXError`	Jumlah GetEventPrediction permintaan di mana Amazon Fraud Detector mengembalikan kode respons HTTP 5xx. Untuk setiap respons 5xx, 1 dikirim. Dimensi yang Benar: `DetectorID`
`Prediction`	Jumlah prediksi. 1 dikirim jika berhasil. Dimensi yang Valid:`DetectorID` , `DetectorVersionID`
`PredictionLatency`	Interval waktu yang dibutuhkan untuk operasi prediksi. Dimensi yang Valid:`DetectorID` , `DetectorVersionID` Satuan: Milidetik
`PredictionError`	Jumlah prediksi di mana Amazon Fraud Detector mengalami kesalahan. 1 dikirim jika terjadi kesalahan . Dimensi yang Valid:`DetectorID` , `DetectorVersionID`

Metrik	Deskripsi
`VariableUsed`	Jumlah GetEventPrediction permintaan di mana variabel digunakan sebagai bagian dari evaluasi. Dimensi yang Valid:`DetectorID` ,`DetectorVersionID` , `VariableName`
`VariableDefaultReturned`	Jumlah GetEventPrediction permintaan di mana variabel tidak hadir sebagai bagian dari Atribut Peristiwa dan oleh karena itu nilai default untuk variabel digunakan selama evaluasi. Dimensi yang Valid:`DetectorID` ,`DetectorVersionID` , `VariableName`
`RuleNotEvaluated`	Jumlah GetEventPrediction permintaan di mana aturan tidak dievaluasi karena aturan sebelumnya cocok. Dimensi yang Valid:`DetectorID` ,`DetectorVersionID` , `RuleID`
`RuleEvaluateTrue`	Jumlah GetEventPrediction permintaan di mana aturan dipicu sebagai Benar dan hasil aturan dikembalikan. Dimensi yang Valid:`DetectorID` ,`DetectorVersionID` , `RuleID`
`RuleEvaluateFalse`	Jumlah GetEventPrediction permintaan di mana aturan dievaluasi ke False. Dimensi yang Valid:`DetectorID` ,`DetectorVersionID` , `RuleID`

Metrik	Deskripsi
RuleEvaluateError	Jumlah GetEventPrediction permintaan di mana aturan mengevaluasi kesalahan Dimensi yang Valid:DetectorID ,DetectorVersionID , RuleID
OutcomeReturned	Jumlah GetEventPrediction panggilan di mana hasil yang ditentukan dikembalikan. Dimensi yang Valid:DetectorID ,DetectorVersionID , OutcomeName
ModelInvocation (Amazon SageMaker model endpoint)	Jumlah GetEventPrediction permintaan di mana titik akhir SageMaker model dipanggil sebagai bagian dari evaluasi. Dimensi yang Valid:DetectorID ,DetectorVersionID , ModelEndpoint
ModelInvocationError (Amazon SageMaker model endpoint)	Jumlah GetEventPrediction permintaan di mana titik akhir SageMaker model yang dipanggil mengembalikan kesalahan selama evaluasi. Dimensi yang Valid:DetectorID ,DetectorVersionID , ModelEndpoint
ModelInvocationLatency (Amazon SageMaker model endpoint)	Interval waktu yang dibutuhkan oleh Model Impor untuk merespons seperti yang dilihat dari Amazon Fraud Detector. Interval ini hanya mencakup pemanggilan model. Dimensi yang Valid:DetectorID ,DetectorVersionID , ModelEndpoint Satuan: Milidetik

Metrik	Deskripsi
`ModelInvocation`	Jumlah GetEventPrediction permintaan di mana model dipanggil sebagai bagian dari evaluasi. Dimensi yang Valid: `DetectorID DetectorVersionID ,,ModelType , ModelID`
`ModelInvocationError`	Jumlah GetEventPrediction permintaan di mana model Amazon Fraud Detector mengembalikan kesalahan selama evaluasi. Dimensi yang Valid: `DetectorID DetectorVersionID ,,ModelType , ModelID`
`ModelInvocationLatency`	Interval waktu yang dibutuhkan oleh Amazon Fraud Detector Model untuk merespons seperti yang dilihat dari Amazon Fraud Detector. Interval ini hanya mencakup pemanggilan model. Dimensi yang Valid: `DetectorID DetectorVersionID ,,ModelType , ModelID` Satuan: Milidetik

Mencatat Panggilan API Fraud Detector Amazon dengan AWS CloudTrail

Amazon Fraud Detector terintegrasi denganAWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di Amazon Fraud Detector. CloudTrail menangkap semua panggilan API untuk Amazon Fraud Detector sebagai peristiwa, termasuk panggilan dari konsol Amazon Fraud Detector dan panggilan dari kode ke Amazon Fraud Detector API.

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara terus menerus ke bucket Amazon S3, termasuk peristiwa untuk Amazon Fraud Detector. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara. Dengan menggunakan informasi yang dikumpulkan CloudTrail, Anda dapat menentukan

permintaan yang dibuat ke Amazon Fraud Detector, alamat IP tempat permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk mempelajari selengkapnya CloudTrail, lihat Panduan AWS CloudTrail Pengguna.

Informasi Fraud Detector Amazon di CloudTrail

CloudTrail diaktifkan di AWS akun Anda saat Anda membuat akun. Ketika aktivitas terjadi di Amazon Fraud Detector, aktivitas tersebut direkam dalam suatu CloudTrail peristiwa bersama dengan peristiwa AWS layanan lainnya dalam riwayat Acara. Anda dapat melihat, mencari, dan mengunduh peristiwa terbaru di akun AWS. Untuk informasi selengkapnya, lihat Melihat Acara dengan Riwayat CloudTrail Acara.

Untuk catatan peristiwa yang sedang berlangsung di AWS akun Anda, termasuk peristiwa untuk Amazon Fraud Detector, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, ketika Anda membuat jejak di konsol tersebut, jejak diterapkan ke semua Wilayah AWS. Jejak mencatat kejadian dari semua Wilayah di partisi AWS dan mengirimkan berkas log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- Gambaran umum untuk Membuat Jejak
- CloudTrail Layanan dan Integrasi yang Didukung
- Mengonfigurasi Notifikasi Amazon SNS untuk CloudTrail
- Menerima File CloudTrail Log dari Beberapa Wilayah dan Menerima File CloudTrail Log dari Beberapa Akun

Amazon Fraud Detector mendukung pencatatan setiap tindakan (operasi API) sebagai peristiwa dalam file CloudTrail log. Untuk informasi selengkapnya, lihat Tindakan.

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

- Baik permintaan tersebut dibuat dengan kredensial pengguna root atau pengguna.
- Baik permintaan tersebut dibuat dengan kredensial keamanan sementara untuk peran atau pengguna gabungan.
- Apakah permintaan dibuat oleh layanan AWS lain.

Untuk informasi lain, lihat Elemen userIdentity CloudTrail .

Memahami Entri File Log Detektor Fraud Detector Amazon

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa mewakili permintaan tunggal dari sumber mana pun dan mencakup informasi tentang operasi yang diminta, tanggal dan waktu operasi, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `GetDetectors` operasi.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "principal-id",
        "arn": "arn:aws:iam::user-arn",
        "accountId": "account-id",
        "accessKeyId": "access-key",
        "userName": "user-name"
    },
    "eventTime": "2019-11-22T02:18:03Z",
    "eventSource": "frauddetector.amazonaws.com",
    "eventName": "GetDetectors",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "source-ip-address",
    "userAgent": "aws-cli/1.11.16 Python/2.7.11 Darwin/15.6.0 botocore/1.4.73",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "request-id",
    "eventID": "event-id",
    "eventType": "AwsApiCall",
    "recipientAccountId": "recipient-account-id"
}
```

Pemecahan Masalah

Bagian berikut membantu Anda memecahkan masalah yang mungkin Anda temui saat bekerja dengan Amazon Fraud Detector

Memecahkan masalah data pelatihan

Gunakan informasi di bagian ini untuk membantu mendiagnosis dan menyelesaikan masalah yang mungkin Anda lihat di panel diagnostik pelatihan Model di konsol Amazon Fraud Detector saat Anda melatih model.

Masalah yang ditampilkan di panel diagnostik pelatihan Model dikategorikan sebagai berikut. Persyaratan untuk mengatasi masalah ini tergantung pada kategori masalah.

- Kesala
  - menyebabkan pelatihan model gagal. Masalah-masalah ini harus diatasi agar model dapat dilatih dengan sukses.
- Pering
  - menyebabkan pelatihan model berlanjut, namun, beberapa variabel mungkin dikecualikan dalam proses pelatihan. Periksa panduan yang relevan di bagian ini untuk meningkatkan kualitas kumpulan data Anda.
- Informa
  (Info) - tidak berdampak pada pelatihan model dan semua variabel digunakan untuk pelatihan. Kami menyarankan Anda memeriksa panduan yang relevan di bagian ini untuk lebih meningkatkan kualitas dataset dan kinerja model Anda.

Topik

- Tipe variabel yang hilang atau salah
- Nilai variabel yang hilang
- Nilai variabel unik tidak mencukupi
- Ekspresi variabel salah
- Entitas unik yang tidak mencukupi

Tingkat penipuan yang tidak stabil dalam kumpulan data yang diberikan

Jenis masalah: Kesalahan

Deskripsi

Tingkat penipuan dalam data yang diberikan terlalu tidak stabil dari waktu ke waktu. Pastikan penipuan dan peristiwa sah Anda diambil sampelnya secara seragam dari waktu ke waktu.

Menyebabkan

Kesalahan ini terjadi jika penipuan dan peristiwa yang sah dalam kumpulan data Anda didistribusikan secara tidak merata dan diambil dari slot waktu yang berbeda. Contoh proses pelatihan model Amazon Fraud Detector dan partisi kumpulan data Anda berdasarkan EVENT_TIMESTAMP. Misalnya, jika kumpulan data Anda terdiri dari peristiwa penipuan yang ditarik dari 6 bulan terakhir, tetapi hanya bulan terakhir peristiwa yang sah yang disertakan, kumpulan data dianggap tidak stabil. Dataset yang tidak stabil dapat menyebabkan bias dalam evaluasi kinerja model.

Solusi

Pastikan untuk memberikan data peristiwa penipuan dan sah dari slot waktu yang sama dan tingkat penipuan tidak berubah secara dramatis dari waktu ke waktu.

Data tidak mencukupi

1. Jenis masalah: Kesalahan

   Deskripsi

   Kurang dari 50 baris diberi label sebagai peristiwa penipuan. Pastikan bahwa peristiwa penipuan dan sah melebihi jumlah minimum 50 dan latih kembali model.

   Menyebabkan

Kesalahan ini terjadi jika kumpulan data Anda memiliki lebih sedikit peristiwa yang diberi label penipuan daripada yang diperlukan untuk pelatihan model. Amazon Fraud Detector memerlukan setidaknya 50 peristiwa penipuan untuk melatih model Anda.

Solusi

Pastikan bahwa dataset Anda mencakup minimal 50 peristiwa penipuan. Anda dapat memastikan ini dengan mencakup periode waktu yang lebih lama, jika diperlukan.

2. Jenis masalah: Kesalahan

   Deskripsi

   Kurang dari 50 baris diberi label sebagai peristiwa yang sah. Pastikan bahwa peristiwa penipuan dan sah melebihi jumlah minimum $threshold dan latih kembali modelnya.

   Menyebabkan

   Kesalahan ini terjadi jika kumpulan data Anda memiliki lebih sedikit peristiwa yang diberi label sah daripada yang diperlukan untuk pelatihan model. Amazon Fraud Detector memerlukan setidaknya 50 peristiwa yang sah untuk melatih model Anda.

   Solusi

   Pastikan kumpulan data Anda menyertakan minimal 50 peristiwa yang sah. Anda dapat memastikan ini dengan mencakup periode waktu yang lebih lama, jika diperlukan.

3. Jenis masalah: Kesalahan

   Deskripsi

   Jumlah entitas unik yang terkait dengan penipuan kurang dari 100. Pertimbangkan untuk memasukkan lebih banyak contoh entitas penipuan untuk meningkatkan kinerja.

   Menyebabkan

   Kesalahan ini terjadi jika kumpulan data Anda memiliki lebih sedikit entitas dengan peristiwa penipuan daripada yang diperlukan untuk pelatihan model. Model Transaction Fraud Insights (TFI) membutuhkan setidaknya 100 entitas dengan peristiwa penipuan untuk memastikan cakupan maksimum ruang penipuan. Model mungkin tidak menggeneralisasi dengan baik jika semua peristiwa penipuan dilakukan oleh sekelompok kecil entitas.

Solusi

Pastikan kumpulan data Anda mencakup setidaknya 100 entitas dengan peristiwa penipuan. Anda dapat memastikan ini mencakup periode waktu yang lebih lama, jika diperlukan.

4. Jenis masalah: Kesalahan

   Deskripsi

   Jumlah entitas unik yang terkait dengan sah kurang dari 100. Pertimbangkan untuk memasukkan lebih banyak contoh entitas yang sah untuk meningkatkan kinerja.

   Menyebabkan

   Kesalahan ini terjadi jika kumpulan data Anda memiliki lebih sedikit entitas dengan peristiwa yang sah daripada yang diperlukan untuk pelatihan model. Model Transaction Fraud Insights (TFI) membutuhkan setidaknya 100 entitas dengan peristiwa yang sah untuk memastikan cakupan maksimum ruang penipuan. Model mungkin tidak menggeneralisasi dengan baik jika semua peristiwa yang sah dilakukan oleh sekelompok kecil entitas.

   Solusi

   Pastikan kumpulan data Anda menyertakan setidaknya 100 entitas dengan peristiwa yang sah. Anda dapat memastikan ini mencakup periode waktu yang lebih lama, jika diperlukan.

5. Jenis masalah: Kesalahan

   Deskripsi

   Kurang dari 100 baris ada dalam kumpulan data. Pastikan ada lebih dari 100 baris dalam kumpulan data total dan setidaknya 50 baris diberi label sebagai penipuan.

   Menyebabkan

   Kesalahan ini terjadi jika kumpulan data Anda berisi kurang dari 100 catatan. Amazon Fraud Detector memerlukan data dari setidaknya 100 peristiwa (catatan) dalam kumpulan data Anda untuk pelatihan model.

   Solusi

   Pastikan Anda memiliki data dari lebih dari 100 peristiwa dalam kumpulan data Anda.

Nilai EVENT_LABEL yang hilang atau berbeda

1. Jenis masalah: Kesalahan

   Deskripsi

   Lebih besar dari 1% kolom EVENT_LABEL Anda adalah nol atau nilai selain yang ditentukan dalam konfigurasi model. **`$label_values`** Pastikan Anda memiliki kurang dari 1% nilai yang hilang di kolom EVENT_LABEL Anda dan nilainya ditentukan dalam konfigurasi model. **`$label_values`**

   Menyebabkan

   Kesalahan ini terjadi karena salah satu alasan berikut:

   - Lebih dari 1% catatan dalam file CSV yang berisi data pelatihan Anda memiliki nilai yang hilang di kolom EVENT_LABEL.
   - Lebih dari 1% catatan dalam file CSV yang berisi data pelatihan Anda memiliki nilai di kolom EVENT_LABEL yang berbeda dari yang terkait dengan jenis acara Anda.

   Model Online Fraud Insights (OFI) mengharuskan kolom EVENT_LABEL di setiap rekaman diisi dengan salah satu label yang terkait dengan jenis acara Anda (atau, dipetakan). `CreateModelVersion`

   Solusi

   Jika kesalahan ini disebabkan oleh nilai EVENT_LABEL yang hilang, pertimbangkan untuk menetapkan label yang tepat ke catatan tersebut atau menghapus catatan tersebut dari kumpulan data Anda. Jika kesalahan ini karena label dari beberapa catatan tidak ada**`label_values`**, pastikan untuk menambahkan semua nilai di kolom EVENT_LABEL ke label jenis acara dan dipetakan ke penipuan atau sah (penipuan, sah) dalam pembuatan model.

2. Jenis masalah: Informasi

   Deskripsi

   Kolom EVENT_LABEL berisi nilai nol atau nilai label selain yang ditentukan dalam konfigurasi model. **`$label_values`** Nilai-nilai yang tidak konsisten ini diubah menjadi 'bukan penipuan' sebelum pelatihan.

   Menyebabkan

Anda mendapatkan informasi ini karena salah satu alasan berikut:

- Kurang dari 1% catatan dalam file CSV yang berisi data pelatihan Anda memiliki nilai yang hilang di kolom EVENT_LABEL
- Kurang dari 1% catatan dalam file CSV yang berisi data pelatihan Anda memiliki nilai di kolom EVENT_LABEL yang berbeda dari yang terkait dengan jenis acara Anda.

Pelatihan model dalam kedua kasus akan berhasil. Namun, nilai label dari peristiwa yang memiliki nilai label yang hilang atau tidak dipetakan diubah menjadi sah. Jika Anda menganggap ini sebagai masalah, ikuti solusi yang disediakan di bawah ini.

Solusi

Jika ada nilai EVENT_LABEL yang hilang dalam kumpulan data Anda, pertimbangkan untuk menghapus catatan tersebut dari kumpulan data Anda. Jika nilai yang diberikan untuk EVENT_LABELS tersebut tidak dipetakan, pastikan bahwa semua nilai tersebut dipetakan ke penipuan atau sah (penipuan, sah) untuk setiap peristiwa.

Nilai EVENT_TIMESTAMP hilang atau salah

1. Jenis masalah: Kesalahan

   Deskripsi

   Kumpulan data pelatihan Anda berisi EVENT_TIMESTAMP dengan stempel waktu yang tidak sesuai dengan format yang diterima. Pastikan formatnya adalah salah satu format tanggal/stempel waktu yang diterima.

   Menyebabkan

   Kesalahan ini terjadi jika kolom EVENT_TIMESTAMP berisi nilai yang tidak sesuai dengan format stempel waktu yang didukung oleh Amazon Fraud Detector.

   Solusi

   Pastikan bahwa nilai yang disediakan untuk kolom EVENT_TIMESTAMP sesuai dengan format stempel waktu yang didukung. Jika Anda memiliki nilai yang hilang di kolom EVENT_TIMESTAMP, Anda dapat mengisi ulang nilai tersebut dengan nilai menggunakan format stempel waktu yang

didukung atau mempertimbangkan untuk menghapus acara sepenuhnya alih-alih memasukkan string seperti,, atau. `none` `null` `missing`

2. Jenis masalah: Kesalahan

   Kumpulan data pelatihan Anda berisi EVENT_TIMESTAMP dengan nilai yang hilang. Pastikan Anda tidak memiliki nilai yang hilang.

   Menyebabkan

   Kesalahan ini terjadi jika kolom EVENT_TIMESTAMP dalam kumpulan data Anda memiliki nilai yang hilang. Amazon Fraud Detector mengharuskan kolom EVENT_TIMESTAMP dalam kumpulan data Anda memiliki nilai.

   Solusi

   Pastikan kolom EVENT_TIMESTAMP dalam kumpulan data Anda memiliki nilai dan nilai tersebut sesuai dengan format stempel waktu yang didukung. Jika Anda memiliki nilai yang hilang di kolom EVENT_TIMESTAMP, Anda dapat mengisi ulang nilai tersebut dengan nilai menggunakan format stempel waktu yang didukung atau mempertimbangkan untuk menghapus acara sepenuhnya alih-alih memasukkan string seperti,, atau. `none` `null` `missing`

Data tidak tertelan

Jenis masalah: Kesalahan

Deskripsi

Tidak ada acara tertelan yang ditemukan untuk pelatihan, silakan periksa konfigurasi pelatihan Anda.

Menyebabkan

Kesalahan ini terjadi jika Anda membuat model dengan data peristiwa yang disimpan dengan Amazon Fraud Detector tetapi tidak mengimpor dataset Anda ke Amazon Fraud Detector sebelum Anda mulai melatih model Anda.

Solusi

Gunakan operasi `SendEvent` API, operasi `CreateBatchImportJob` API, atau fitur impor batch di konsol Amazon Fraud Detector, untuk mengimpor data peristiwa terlebih dahulu, lalu melatih model Anda. Lihat Kumpulan data peristiwa tersimpan untuk informasi selengkapnya.

> **Note**
>
> Kami sarankan menunggu 10 menit setelah Anda selesai mengimpor data Anda sebelum menggunakannya untuk melatih model Anda.

Anda dapat menggunakan konsol Amazon Fraud Detector untuk memeriksa jumlah peristiwa yang sudah disimpan untuk setiap jenis acara. Lihat Melihat metrik peristiwa yang disimpan untuk informasi selengkapnya.

Variabel tidak mencukupi

Jenis masalah: Kesalahan

Deskripsi

Dataset harus berisi setidaknya 2 variabel yang cocok untuk pelatihan.

Menyebabkan

Kesalahan ini terjadi jika kumpulan data Anda berisi kurang dari 2 variabel yang cocok untuk pelatihan model. Amazon Fraud Detector menganggap variabel yang cocok untuk pelatihan model hanya jika melewati semua validasi. Jika variabel gagal validasi, itu dikecualikan dalam pelatihan model dan Anda akan melihat pesan di Diagnostik pelatihan Model.

Solusi

Pastikan kumpulan data Anda memiliki setidaknya dua variabel yang diisi dengan nilai dan lulus semua validasi data. Perhatikan bahwa baris metadata peristiwa di mana Anda telah memberikan header kolom Anda (EVENT_TIMESTAMP, EVENT_ID, ENTITY_ID, EVENT_LABEL, dll.) tidak dianggap sebagai variabel.

Tipe variabel yang hilang atau salah

Jenis masalah: Peringatan

Deskripsi

Tipe data yang diharapkan untuk **`$variable_name`** adalah NUMERIK. Tinjau dan perbarui **`$variable_name`** dalam kumpulan data Anda dan latih kembali modelnya.

Menyebabkan

Anda mendapatkan peringatan ini jika variabel didefinisikan sebagai variabel NUMERIK, tetapi dalam kumpulan data, ia memiliki nilai yang tidak dapat dikonversi ke NUMERIC. Akibatnya, variabel itu dikecualikan dalam pelatihan model.

Solusi

Jika Anda ingin menyimpannya sebagai variabel NUMERIK, pastikan bahwa nilai yang Anda berikan dapat dikonversi ke nomor float. Perhatikan bahwa jika variabel berisi nilai yang hilang, jangan mengisinya dengan string sepertinonene,`null`, atau`missing`. Jika variabel memang berisi nilai non-numerik, buat ulang sebagai tipe variabel CATEGORICAL atau FREE_FORM_TEXT.

Nilai variabel yang hilang

Jenis masalah: Peringatan

Deskripsi

Lebih besar dari **`$threshold`** nilai untuk **`$variable_name`** hilang dari kumpulan data pelatihan Anda. Pertimbangkan untuk memodifikasi **`$variable_name`** dalam kumpulan data dan pelatihan ulang Anda untuk meningkatkan kinerja.

Menyebabkan

Anda mendapatkan peringatan ini jika variabel yang ditentukan dijatuhkan karena terlalu banyak nilai yang hilang. Amazon Fraud Detector memungkinkan nilai yang hilang untuk variabel. Namun, jika satu variabel memiliki terlalu banyak nilai yang hilang, itu tidak berkontribusi banyak pada model dan variabel itu dijatuhkan dalam pelatihan model.

Solusi

Pertama, verifikasi bahwa nilai yang hilang itu bukan karena kesalahan dalam pengumpulan dan persiapan data. Jika itu kesalahan, maka Anda dapat mempertimbangkan untuk menjatuhkannya dari pelatihan model Anda. Namun, jika Anda yakin nilai-nilai yang hilang itu berharga dan masih ingin mempertahankan variabel itu, Anda dapat secara manual mengisi nilai yang hilang dengan konstanta dalam pelatihan model dan inferensi waktu nyata.

Nilai variabel unik tidak mencukupi

Jenis masalah: Peringatan

Deskripsi

Hitungan nilai unik **$variable_name** lebih rendah dari 100. Tinjau dan perbarui **$variable_name** dalam kumpulan data Anda dan latih kembali modelnya.

Menyebabkan

Anda mendapatkan peringatan ini jika jumlah nilai unik dari variabel yang ditentukan kurang dari 100. Ambang batas berbeda tergantung pada jenis variabel. Dengan nilai unik yang sangat sedikit, ada risiko bahwa kumpulan data tidak cukup umum untuk mencakup ruang fitur variabel itu. Akibatnya, model mungkin tidak menggeneralisasi dengan baik pada prediksi waktu nyata.

Solusi

Pertama, pastikan distribusi variabel mewakili lalu lintas bisnis nyata. Kemudian, Anda dapat mengadopsi lebih banyak variabel terlatih dengan kardinalitas yang lebih tinggi, seperti menggunakan `full_customer_name` alih-alih `first_name` dan `last_name` secara terpisah atau mengubah tipe variabel menjadi CATEGORICAL, yang memungkinkan kardinalitas lebih rendah.

Ekspresi variabel salah

1. Jenis masalah: Informasi

   Deskripsi

   Lebih dari 50% **$email_variable_name** nilai tidak cocok dengan ekspresi reguler yang diharapkan http://emailregex.com. Pertimbangkan untuk memodifikasi **$email_variable_name** dalam kumpulan data dan pelatihan ulang Anda untuk meningkatkan kinerja.

   Menyebabkan

   Informasi ini ditampilkan jika lebih dari 50% catatan dalam dataset Anda memiliki nilai email yang tidak sesuai dengan ekspresi email biasa dan karena itu gagal validasi.

   Solusi

   Format nilai variabel email agar sesuai dengan ekspresi reguler. Jika ada nilai email yang hilang, kami sarankan untuk membiarkannya kosong alih-alih mengisinya dengan string sepertinone,`null`, atau`missing`.
2. Jenis masalah: Informasi

   Deskripsi

Lebih dari 50% **$IP_variable_name** nilai tidak cocok dengan ekspresi reguler untuk alamat IPv4 atau IPv6 https://digitalfortress.tech/tricks/top-15 - /. commonly-used-regex Pertimbangkan untuk memodifikasi **$IP_variable_name** dalam kumpulan data dan pelatihan ulang Anda untuk meningkatkan kinerja.

Menyebabkan

Informasi ini ditampilkan jika lebih dari 50% catatan dalam dataset Anda memiliki nilai IP yang tidak sesuai dengan ekspresi IP reguler dan karena itu gagal validasi.

Solusi

Format nilai IP agar sesuai dengan ekspresi reguler. Jika ada nilai IP yang hilang, kami sarankan untuk membiarkannya kosong alih-alih mengisinya dengan string sepertinone,`null`, atau`missing`.

3. Jenis masalah: Informasi

   Deskripsi

   Lebih dari 50% **$phone_variable_name** nilai tidak cocok dengan ekspresi reguler telepon dasar /$pattern/. Pertimbangkan untuk memodifikasi **$phone_variable_name** dalam kumpulan data dan pelatihan ulang Anda untuk meningkatkan kinerja.

   Menyebabkan

   Informasi ini ditampilkan jika lebih dari 50% catatan dalam dataset Anda memiliki nomor telepon yang tidak sesuai dengan ekspresi nomor telepon biasa dan karena itu gagal validasi.

   Solusi

   Format nomor telepon agar sesuai dengan ekspresi reguler. Jika ada nomor telepon yang hilang, kami sarankan untuk membiarkannya kosong daripada mengisinya dengan string sepertinone,`null`, atau`missing`.

Entitas unik yang tidak mencukupi

Jenis masalah: Informasi

Deskripsi

Jumlah entitas unik kurang dari 1500. Pertimbangkan untuk memasukkan lebih banyak data untuk meningkatkan kinerja.

Menyebabkan

Informasi ini ditampilkan jika kumpulan data Anda memiliki jumlah entitas unik yang lebih kecil daripada nomor yang disarankan. Model Transaction Fraud Insights (TFI) menggunakan agregat deret waktu dan fitur transaksi generik untuk memberikan kinerja terbaik. Jika kumpulan data Anda memiliki terlalu sedikit entitas unik, maka sebagian besar data generik Anda seperti IP_ADDRESS, EMAIL_ADDRESS, mungkin tidak memiliki nilai unik. Kemudian, ada juga risiko bahwa kumpulan data ini tidak cukup umum untuk mencakup ruang fitur variabel itu. Akibatnya, model mungkin tidak menggeneralisasi dengan baik pada transaksi dari entitas baru yang baru.

Solusi

Sertakan lebih banyak entitas. Perpanjang rentang waktu data pelatihan Anda, jika diperlukan.

Quotas

Akun AndaAkun AWS memiliki kuota default, yang sebelumnya disebut sebagai batasan, untuk setiap layanan Amazon Web Service. Kecuali dinyatakan lain, setiap kuota bersifat khusus per Wilayah. Akun Anda dapat meminta peningkatan kuota untuk semua kuota yang dapat disesuaikan yang disebutkan dalam tabel di bawah. Untuk informasi selengkapnya, lihat Permintaan peningkatan kuota

Tabel berikut menguraikan kuota Amazon Fraud Detector berdasarkan komponen.

Model Amazon Fraud Detector

Nama kuota	Kuota default	Dapat Disesuaikan
Ukuran data pelatihan	5 GB	Tidak
Model per akun	50	Tidak
Versi per model	200	Tidak
Versi model yang digunakan per akun	5	Tidak
tugas pelatihan serentak per akun	3	Tidak
Pekerjaan pelatihan bersamaan per model	1	Tidak

Detektor/variabel/hasil/aturan Amazon Fraud Detector

Nama kuota	Kuota default	Dapat Disesuaikan
Variabel per akun	5000	Tidak
Aturan per akun	5000	Tidak

Nama kuota	Kuota default	Dapat Disesuaikan
Daftar per aturan	3	Tidak
Hasil per akun	5000	Tidak
Detektor per akun	100	Tidak
Daftar per detektor	30	Tidak
Versi draf per detektor	100	Tidak
Model per versi detektor	10	Tidak
Label per akun	100	Tidak
Jenis acara per akun	100	Tidak
Jenis ententitas per akun	100	Tidak

Amazon Fraud Detector API

Nama kuota	Kuota default	Dapat Disesuaikan
GetEventPrediction Panggilan API per detik	200 TPS	Ya
Ukuran payload per panggilan GetEventPrediction API	256 KB	Tidak
Jumlah input per panggilan GetEventPrediction API	5000	Tidak

Riwayat dokumen

Tabel berikut menjelaskan perubahan penting dalam Panduan Pengguna Amazon Fraud Detector. Kami juga sering memperbarui Panduan Pengguna Amazon Fraud Detector untuk mengatasi umpan balik yang Anda kirimkan kepada kami.

Perubahan	Deskripsi	Tanggal
Variabel baru dan tipe data	Amazon Fraud Detector memperkenalkan jenis variabel baru dan tipe data yang dapat Anda gunakan untuk mengekstrak informasi yang berguna.	5 Juni 2023
Orkestrasi acara	Orkestrasi Acara memudahkan Anda mengirim acara Layanan AWS untuk pemrosesan hilir, menggunakan Amazon. EventBridge	30 Mei 2023
Daftar	Sumber daya Daftar memungkinkan Anda untuk mereferensikan serangkaian nilai seperti alamat IP atau alamat email, sebagai bagian dari aturan. Gunakan daftar dalam aturan untuk mengizinkan atau menolak akses atau transaksi.	Februari 14, 2023
Data Model Explorer	Data Models Explorer memberikan wawasan tentang elemen data yang diperlukan oleh Amazon Fraud Detector untuk membuat model deteksi penipuan Anda. Gunakan data	15 Desember 2022

	model explorer sebelum Anda mempersiapkan dataset acara Anda.	
Model Wawasan Pengambil alihan Akun	Gunakan model Account takeover insights (ATI) untuk mendeteksi akun yang disusupi melalui pengambil alihan berbahaya, phishing, atau dari kredensi yang dicuri.	21 Juli 2022
Pembaruan Bab	Memperbarui babak pengantar dengan informasi tambahan tentang Amazon Fraud Detector	April 11, 2022
Pengayaan variabel	Aktifkan pengayaan beberapa data mentah yang Anda berikan untuk meningkat kan kinerja model yang menggunakan elemen data ini dan yang dilatih sebelum 8 Februari 2022.	Februari 8, 2022
Kebijakan opt-out	Gunakan kebijakan penyisihan untuk tidak menggunakan data acara Anda untuk mengemban gkan atau meningkatkan kualitas Amazon Fraud Detector.	Januari 6, 2022

Bingung wakil pencegahan	Buat kebijakan untuk mencegah pihak ketiga atau entitas lintas layanan memanipulasi entitas dengan izin untuk bertindak atas namanya guna mendapatkan akses ke sumber daya di akun Anda.	Desember 6, 2021
Buat dataset acara	Gunakan panduan yang disediakan di Buat kumpulan data acara untuk menyiapkan dan mengumpulkan data untuk melatih model Anda.	November 22, 2021
Penjelasan prediksi	Gunakan penjelasan Prediksi untuk mendapatkan wawasan tentang bagaimana setiap variabel peristiwa memengaruhi skor prediksi penipuan model Anda.	November 10, 2021
Memecahkan masalah	Gunakan informasi dalam Memecahkan masalah data latihan untuk membantu mendiagnosis dan menyelesaikan masalah yang mungkin Anda lihat di konsol Amazon Fraud Detector saat Anda melatih model.	Oktober 11, 2021
Model wawasan penipuan transaksi	Gunakan model Transaction fraud insights (TFI) untuk mendeteksi penipuan online atau card-not-present transaksi.	Oktober 11, 2021

Acara yang disimpan	Simpan data peristiwa Anda di Amazon Fraud Detector dan gunakan data yang tersimpan untuk kemudian melatih model Anda. Dengan menyimpan data peristiwa di Amazon Fraud Detector, Anda dapat melatih model yang menggunakan variabel komputasi otomatis untuk meningkatkan kinerja, menyederhanakan pelatihan ulang model, dan memperbar ui label penipuan untuk menutup loop umpan balik machine learning.	Oktober 11, 2021
Model variabel pentingnya	Gunakan Model variabel pentingnya untuk mendapatk an wawasan tentang apa yang mendorong kinerja model Anda naik atau turun dan mana dari variabel model Anda berkontribusi paling. Dan kemudian tweak model Anda untuk meningkatkan kinerja secara keseluruhan.	Juli 9, 2021
Integrasi dengan AWS CloudFormation	Gunakan AWS CloudForm ation untuk mengelola sumber daya Amazon Fraud Detector Anda.	10 Mei 2021

Prediksi batch	Gunakan prediksi Batch untuk mendapatkan prediksi untuk serangkaian peristiwa yang tidak memerlukan penilaian waktu nyata.	31 Maret 2021
Pengerjaan ulang Bab	Pengerjaan ulang Memulai dan bagian lainnya	17 Juli 2020
Rilis awal	Rilis pertama	2 Desember 2019

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