



デベロッパーガイド

Amazon Comprehend Medical



Amazon Comprehend Medical: デベロッパーガイド

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスは Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

Amazon Comprehend Medical とは	1
重要な注意点	1
Amazon Comprehend Medical のユースケース	1
Amazon Comprehend Medical の利点	2
HIPAA への準拠	3
Amazon Comprehend Medical へのアクセス	3
Amazon Comprehend Medical の開始方法	4
仕組み	5
同期エンティティ検出	5
非同期バッチ分析	5
オントロジーリンク	6
病状の ICD-10-CM ナレッジベース内の概念へのリンク	6
薬剤の RxNorm ナレッジベース内の概念へのリンク	6
医療概念の SNOMED CT ナレッジベース内の概念へのリンク	6
はじめに	8
ステップ 1: アカウントを設定する	8
AWS へのサインアップ	8
IAM ユーザーの作成	9
次のステップ	10
ステップ 2: をセットアップする AWS CLI	10
次のステップ	11
ステップ 3: コンソールの使用開始	11
コンソールを使用した臨床テキストの分析	11
ステップ 4: API の使用開始	14
AWS Command Line Interfaceを使用した医療エンティティの検出	14
AWS SDK for Javaを使用した医療エンティティの検出	16
AWS SDK for Python (Boto)を使用した医療エンティティの検出	17
VPC エンドポイント (AWS PrivateLink)	19
Amazon Comprehend Medical VPC エンドポイントに関する考慮事項	19
Amazon Comprehend Medical のインターフェイス VPC エンドポイントの作成	19
Amazon Comprehend Medical の VPC エンドポイントポリシーの作成	20
テキスト分析 API	22
エンティティの検出 (バージョン 2)	22
人体カテゴリ	24

行動、環境、社会的健康のカテゴリ	24
病状カテゴリ	25
薬剤カテゴリ	26
保護医療情報カテゴリ	27
検査、治療、処置のカテゴリ	27
時間表現のカテゴリ	28
PHI の検出	29
テキスト分析バッチ API	34
重要な注意点	1
API を使用したバッチ分析の実行	34
コンソールを使用したバッチ分析の実行	35
IAM ポリシー	35
バッチ分析の出カフファイル	37
オントロジーリンク API	40
InferICD10CM	40
ICD-10-CM のカテゴリ	40
ICD-10-CM のタイプ	40
ICD-10-CM の特性	41
ICD-10-CM の属性	41
時間表現のカテゴリ	41
型	42
関係タイプ	42
入力とレスポンスの例	42
InferRxNorm	45
重要な注意点	1
RxNorm のカテゴリ	45
RxNorm のタイプ	46
RxNorm の属性	46
RxNorm の特性	46
入力とレスポンスの例	46
InferSNOMEDCT	49
人体カテゴリ	50
病状カテゴリ	50
検査、治療、処置のカテゴリ	51
SNOMED CT の詳細	52
入力とレスポンスの例	52

オントロジーリンクのバッチ分析	60
バッチ分析の実行	60
IAM ポリシー	61
バッチ分析の出カファイル	63
セキュリティ	66
データ保護	66
認証とアクセスコントロール	68
認証	68
アクセスコントロール	69
アクセス管理の概要	69
Amazon Comprehend Medical でのアイデンティティベースのポリシー (IAM ポリシー) の使 用	72
Amazon Comprehend Medical API のアクセス許可リファレンス	81
AWS マネージドポリシー	81
を使用した Amazon Comprehend Medical API コールのログ記録 AWS CloudTrail	83
CloudTrail での Amazon Comprehend Medical 情報	83
Amazon Comprehend Medical ログファイルエントリについて	84
コンプライアンス検証	86
耐障害性	86
インフラストラクチャセキュリティ	87
ガイドラインとクォータ	88
重要な注意点	88
サポートされるリージョン	88
スロットリング	88
全体的なクォータ	88
ドキュメント履歴	97
Amazon Comprehend Medical の変更ログ	107
InferICD10CM の API バージョンの更新	108
InferRxNorm の API バージョンの更新	108
InferSNOMEDCT の API バージョンの更新	108
InferICD10CM の API バージョンの更新	108
InferRxNorm の API バージョンの更新	108
InferSNOMEDCT の API バージョンの更新	109
DetectEntities の API バージョンの更新	109
DetectEntitiesV2 の API バージョンの更新	109
InferRxNorm の API バージョンの更新	109

InferSNOMEDCT の API バージョンの更新	109
InferICD10CM の API バージョンの更新	110
Amazon Comprehend Medical API オペレーションの変更	110
Amazon Comprehend Medical API オペレーションの変更	110
Amazon Comprehend Medical API オペレーションの変更	110
Amazon Comprehend Medical API オペレーションの変更	111
Amazon Comprehend Medical API オペレーションの変更	111
Amazon Comprehend Medical API オペレーションの変更	111
Amazon Comprehend Medical API オペレーションの変更	112
Amazon Comprehend Medical API オペレーションの変更	112
Amazon Comprehend Medical InferSNOMEDCT API オペレーションの変更	112
Amazon Comprehend Medical API オペレーションの変更	113

..... cxviii

Amazon Comprehend Medical とは

Amazon Comprehend Medical は、構造化されていない臨床テキスト (医師の記録、退院要約、検査結果、症例記録など) に含まれた有益な情報を検出して返します。Amazon Comprehend Medical は、自然言語処理 (NLP) モデルを使用して、病状、薬剤、保護医療情報 (PHI) などの医療情報へのテキスト参照であるエンティティを検出します。検出されるエンティティの完全なリストについては、「[エンティティの検出 \(バージョン 2\)](#)」を参照してください。Amazon Comprehend Medical では、これらの検出されたエンティティを、オントロジーリンクオペレーションを通じて RxNorm や ICD-10-CM などの標準化された医療ナレッジベースにリンクすることもできます。

この開発者ガイドの情報は、アプリケーション開発者を対象にしています。このガイドでは、AWS CLI または Amazon Comprehend Medical API を使用して Amazon Comprehend Medical をプログラムで使用する方法について説明します。

Amazon Comprehend Medical の料金は、Amazon Comprehend の料金とは異なります。詳細については、「[Amazon Comprehend Medical の料金](#)」を参照してください。

サポートされている言語

Amazon Comprehend Medical は、英語 (US-EN) のテキストでの医療エンティティのみを検出します。

重要な注意点

Amazon Comprehend Medical は専門家による医療の助言、診断、治療の代用品ではありません。Amazon Comprehend Medical は、検出されたエンティティの精度に対する信頼度を示す信頼スコアを提供します。ユースケースに適した信頼しきい値を特定し、高い精度を必要とする状況では高い信頼しきい値を使用してください。特定のユースケースでは、適切な訓練を受けたレビュー担当者によって人的に結果を見直し、検証する必要があります。例えば、Amazon Comprehend Medical は、訓練を受けた医療専門家による正確さと健全な医療判断の確認後、患者ケアのシナリオでのみ使用してください。

Amazon Comprehend Medical のユースケース

Amazon Comprehend Medical は、以下のヘルスケア用途に使用できます。

- 患者ケース管理と結果: 医師や医療提供者は、従来の形態に適合しない医療情報を管理し、簡単にアクセスできます。患者は、標準的なフォーマットよりも多くの情報を含む説明文で健康上の懸念

を報告できます。医療提供者は、症例記録を分析することで、病状の早期スクリーニングの候補者を治療が困難かつ高額になる前に特定できます。

- **臨床研究** — ライフサイエンスおよび研究機関は、患者を臨床試験に登録するためのマッチングプロセスを最適化できます。Amazon Comprehend Medical を使用して臨床テキスト内の関連情報を検出することにより、研究者はファーマコビジランスを改善し、市販後調査を実施して薬物有害事象を監視できます。また、追跡記録やその他の臨床テキスト内で重要な情報を簡単に検出し、治療の有効性を評価することができます。例えば、患者の説明を分析することにより、特定の治療法に患者がどのように反応するかをより簡単かつ効果的に監視できます。
- **医療費請求および医療収益サイクルの管理** — 支払者は分析を拡張して、臨床記録などの構造化されていないドキュメントを含めることができます。診断に関する詳細情報を分析し、構造化されていないドキュメントから適切な請求コードを判断するために使用できます。自然言語処理 (NLP) は、コンピュータ支援コーディング (CAC) の最も重要なコンポーネントです。Amazon Comprehend Medical は、NLP の最新のテクノロジーを使用して臨床テキストを分析するため、収益までの期間を短縮し、報酬の精度を向上するのに役立ちます。
- **オントロジーリンク** — オントロジーリンク機能を使用して、臨床テキストからエンティティを検出し、それらのエンティティを一般的な医療オントロジーの標準化された概念にリンクします。InferICD10CM は、可能性のある病状をエンティティとして識別します。InferICD10CM は、これらのエンティティを 2021 年版の [国際疾病分類 第 10 回改訂版の臨床改定 \(ICD-10-CM\)](#) の一意のコードにリンクします。InferRxNorm は、臨床テキストに記載されている薬剤をエンティティとして識別し、それらのエンティティを [米国国立医学図書館の RxNorm データベース](#) の正規化された概念識別子にリンクします。InferSNOMEDCT は、病状や解剖学、医学的検査、治療、処置などの医療概念をエンティティとして識別し、[Systematized Nomenclature of Medicine, Clinical Terms \(SNOMED CT\)](#) オントロジーのコードにリンクします。

Amazon Comprehend Medical の利点

Amazon Comprehend Medical を使用する利点には、以下のようなものがあります。

- **アプリケーションへの自然言語処理の簡単かつ強力な統合** — API を使用してアプリケーションにテキスト分析機能を組み込み、強力で正確な自然言語処理を実現します。
- **正確性** — 深層学習テクノロジーを使用してテキストを正確に分析します。精度を向上させるために、複数のドメインにわたって新しいデータを使用して常にモデルがトレーニングされます。
- **スケーラビリティ** — 複数のドキュメントから情報を検出できるため、患者の健康とケアに関して迅速にインサイトを得ることができます。

- AWS の他のサービスとの統合 – Amazon Comprehend Medical は、Amazon S3 や AWS Lambda などの AWS の他のサービスとシームレスに連携するように設計されています。ドキュメントを Amazon S3 に保存したり、Firehose でリアルタイムデータを分析したり、Amazon Transcribe を使用して患者の説明文を Amazon Comprehend Medical で分析できるテキストに書き起こしたりします。AWS Identity and Access Management (IAM) のサポートにより、Amazon Comprehend Medical オペレーションへのアクセスを安全に制御できます。IAM を使用して、AWS ユーザーやグループを作成および管理し、デベロッパーやエンドユーザーに適切なアクセス権を付与できます。
- 低コスト — 分析したドキュメントについてのみ支払いが発生します。最低料金や前払いの義務はありません。

HIPAA への準拠

これは HIPAA 対象サービスです。1996 年米国医療保険の相互運用性と説明責任に関する法律 (HIPAA)、および AWS サービスを使用して保護医療情報 (PHI) を処理、保存、および送信する方法の詳細については、[「HIPAA の概要」](#)を参照してください。

PHI が含まれている Amazon Comprehend Medical への接続は暗号化する必要があります。デフォルトでは、Amazon Comprehend Medical へのすべての接続で HTTPS over TLS を使用します。Amazon Comprehend Medical では、お客様のコンテンツが永続的に保存されることはありません。そのため、サービス内で保管中のデータの暗号化を設定する必要はありません。

Amazon Comprehend Medical へのアクセス

1. AWS マネジメントコンソール — Amazon Comprehend Medical へのアクセスに使用するウェブインターフェイスが用意されています。
2. AWS Command Line Interface (AWS CLI) — Amazon Comprehend Medical を含むさまざまな AWS のサービス用のコマンドが用意されており、Windows、macOS、および Linux でサポートされています。AWS CLI のインストールの詳細については、AWS コマンドラインインターフェイスを参照してください。
3. AWS SDK - AWS には、さまざまなプログラミング言語およびプラットフォーム (Java、Python、Ruby、.NET、iOS、Android など) のライブラリとサンプルコードで構成された SDK (ソフトウェア開発キット) が用意されています。SDK は、Amazon Comprehend Medical および AWS へのプログラムによるアクセスを作成するのに便利です。詳細については、AWS SDK を参照してください。

Amazon Comprehend Medical の開始方法

Amazon Comprehend Medical を初めて使用する場合は、以下のセクションを順に読むことをお勧めします。

1. [Amazon Comprehend Medical の仕組み](#) — このセクションでは、Amazon Comprehend Medical の概念について説明しています。
2. [Amazon Comprehend Medical の開始方法](#) — このセクションでは、アカウントを設定し Amazon Comprehend Medical をテストする方法について説明しています。

Amazon Comprehend Medical の仕組み

Amazon Comprehend Medical は、トレーニング済みの自然言語処理 (NLP) モデルを使用し、エンティティの検出を通じて、構造化されていない臨床テキストを分析します。エンティティは、病状、薬剤、保護医療情報 (PHI) などの医療情報へのテキスト参照です。一部のオペレーションでは、エンティティを検出し、それらのエンティティを標準化されたオントロジーにリンクすることにより、さらに一歩踏み込みます。このモデルは、大量の医療テキストで継続的にトレーニングされるため、トレーニングデータを提供する必要はありません。すべての結果に、検出されたエンティティの精度に対して Amazon Comprehend Medical が持つ信頼度を示す信頼スコアが含まれます。

エンティティの検出およびオントロジーのリンクは、同期オペレーションまたは非同期オペレーションで実行できます。

- 同期オペレーション: 単一のドキュメントで分析を実行でき、分析結果をアプリケーションに直接返します。単一ドキュメントオペレーションは、一度に 1 つのドキュメントで動作するインタラクティブなアプリケーションを作成するときを使用します。
- 非同期オペレーション — Amazon S3 バケットに保存されているドキュメントのコレクションまたはバッチでの分析を実行できます。分析結果は S3 バケットに返されます。

Note

Amazon Comprehend Medical は、英語 (US-EN) のテキストのみを分析できます。

同期エンティティ検出

DetectEntitiesV2 オペレーションおよび DetectPHI オペレーションは、個々の文書から構造化されていない臨床テキスト内のエンティティを検出します。Amazon Comprehend Medical サービスにドキュメントを送信すると、応答で分析結果を受け取ります。

非同期バッチ分析

StartEntitiesDetectionV2Job オペレーションおよび StartPHIDetectionJob オペレーションは、Amazon S3 バケットに保存されている病状、治療、検査と結果、保護健康情報などの医療情報への参照を検出する、非同期ジョブを開始します。検出ジョブの出力は別の Amazon S3 バケットに書き込まれ、そこからさらなる処理やダウンストリーム分析に使用できます。

StartICD10CMInferenceJob オペレーションおよび StartRxNormInferenceJob オペレーションは、エンティティを検出してこれらのエンティティを RxNorm ナレッジベースおよび ICD-10-CM ナレッジベース内の標準化されたコードにリンクする、オントロジーリンクのバッチオペレーションを開始します。

オントロジーリンク

InferICD10CM オペレーション、InferSNOMEDCT オペレーション、および InferRxNorm オペレーションは、可能性のある病状および薬剤を検出し、それらをそれぞれ ICD-10-CM、SNOMED CT、および RxNorm の各ナレッジベース内のコードにリンクします。オントロジーリンクのバッチ分析を使用して、ドキュメントのコレクションまたは単一の大きいドキュメントを分析できます。コンソール、またはオントロジーリンクバッチ API を使用して、継続中のバッチ分析ジョブの開始、停止、リスト表示、および説明を行うオペレーションを実行できます。

病状の ICD-10-CM ナレッジベース内の概念へのリンク

InferICD10CM オペレーションは、可能性のある病状を検出し、それらの病状を国際疾病分類の 2019 年版である第 10 回改訂版の Clinical Modification (ICD-10-CM) のコードにリンクします。検出された可能性のある病状ごとに、Amazon Comprehend Medical は一致する ICD-10-CM コードと説明をリストします。結果内のリストされた病状には、結果内の一致した概念に対するエンティティの精度に関して Amazon Comprehend Medical が持っている信頼度を示す信頼スコアが含まれます。

薬剤の RxNorm ナレッジベース内の概念へのリンク

InferRxNorm オペレーションは、患者記録にリストされている薬剤をエンティティとして識別します。このオペレーションは、エンティティを米国国立医学図書館の RxNorm データベースの概念識別子 (RxCUI) にリンクします。各 RxCUI は、さまざまな強度と処方に対して一意です。結果内のリストされた薬剤には、RxNorm ナレッジベースの概念に一致したエンティティの精度に関して Amazon Comprehend Medical が持っている信頼度を示す信頼スコアが含まれます。Amazon Comprehend Medical は、信頼スコアに基づいて、検出された各薬剤について一致する可能性のある上位の RxCUI を降順にリストします。

医療概念の SNOMED CT ナレッジベース内の概念へのリンク

InferSNOMEDCT オペレーションは、考えられる医療概念をエンティティとして識別し、2021-03 バージョンの「Systematized Nomenclature of Medicine, Clinical Terms (SNOMED CT)」のコードにリンクします。SNOMED CTは、病状や解剖学のほか、医学的検査、治療、処置など、医療概念の包括的な語彙を提供します。Amazon Comprehend Medical は、一致する概念 ID ごとに、上位 5 つの

医療概念を返します。各概念には、信頼スコアと、特性や属性などのコンテキスト情報が含まれます。SNOMED CT の概念 ID を SNOMED CT 多重階層と併用すると、医療事務、レポート、または臨床分析用に患者の臨床データの構造化できます。

Amazon Comprehend Medical の開始方法

Amazon Comprehend Medical の使用を開始するには、AWS アカウントをセットアップし、AWS Identity and Access Management (IAM) ユーザーを作成します。Amazon Comprehend Medical CLI を使用するには、これをダウンロードして設定します。

トピック

- [ステップ 1: AWS アカウントを設定して管理者ユーザーを作成する](#)
- [ステップ 2: AWS Command Line Interface \(AWS CLI\) を設定する](#)
- [ステップ 3: Amazon Comprehend Medical コンソールの使用を開始する](#)
- [ステップ 4: Amazon Comprehend Medical API の使用を開始する](#)

ステップ 1: AWS アカウントを設定して管理者ユーザーを作成する

Amazon Comprehend Medical を初めて使用する場合は、事前に以下のタスクをすべて実行してください。

1. [AWS へのサインアップ](#)
2. [IAM ユーザーの作成](#)

AWS へのサインアップ

アマゾン ウェブ サービス (AWS) にサインアップすると、Amazon Comprehend Medical を含む AWS のすべてのサービスに AWS アカウントが自動的にサインアップされます。料金は、使用するサービスの料金のみが請求されます。

Amazon Comprehend Medical では、使用したリソースに対してのみ料金を支払います。AWS の新規のお客様の場合、無料で Amazon Comprehend Medical の使用を開始できます。詳細については、「[AWS 無料利用枠](#)」を参照してください。

すでに AWS アカウントをお持ちの場合は、次の手順に進んでください。

AWS アカウントを作成するには

1. <https://portal.aws.amazon.com/billing/signup> を開きます。

2. オンラインの手順に従います。

サインアップ手順の一環として、電話またはテキストメッセージを受け取り、電話キーパッドで検証コードを入力します。

にサインアップすると AWS アカウント、AWS アカウントのルートユーザー が作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティベストプラクティスとして、ユーザーに管理アクセス権を割り当て、[ルートユーザーアクセスが必要なタスク](#)の実行にはルートユーザーのみを使用するようにしてください。

次のタスクで AWS アカウント ID が必要となるので、記録しておいてください。

IAM ユーザーの作成

Amazon Comprehend Medical などの AWS のサービスでは、サービスにアクセスする際に認証情報を提供する必要があります。これにより、サービスのリソースへアクセスするための権限があるかどうか判断されます。

AWS アカウントの認証情報ではなく、AWS Identity and Access Management (IAM) を使用して AWS にアクセスすることを強くお勧めします。IAM を使用して AWS にアクセスするには、IAM ユーザーを作成し、管理アクセス許可を持つ IAM グループに追加してから、その IAM ユーザーに管理アクセス許可を付与します。これで、特別な URL と IAM ユーザーの認証情報を使用して AWS にアクセスできます。

このガイドの「使用開始」実習では、管理者権限を持つユーザー (adminuser) が存在すること想定しています。

管理者を作成し、コンソールにサインインするには

1. adminuser という名前のユーザーを AWS アカウントに作成します。手順については、IAM ユーザーガイドの「[最初の IAM ユーザーと管理者グループの作成](#)」を参照してください。
2. 特別な URL AWS マネジメントコンソール を使用して にサインインします。詳細については、[IAM ユーザーガイド](#)のユーザーがアカウントにサインインする方法を参照してください。

IAM の詳細については、以下を参照してください。

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

- [IAM の開始方法](#)
- [IAM ユーザーガイド](#)

次のステップ

[ステップ 2: AWS Command Line Interface \(AWS CLI\) を設定する](#)

ステップ 2: AWS Command Line Interface (AWS CLI) を設定する

「開始方法」の演習のステップを実行する AWS CLI ためには必要ありません。ただし、このガイドの他の実習では必要になります。必要に応じて、このステップをスキップしてに移動し[ステップ 3: Amazon Comprehend Medical コンソールの使用を開始する](#)、AWS CLI 後で をセットアップできます。

を設定するには AWS CLI

1. AWS CLIをダウンロードして設定します。手順については、AWS Command Line Interface ユーザーガイドの次のトピックを参照してください。
 - [のセットアップ AWS Command Line Interface](#)
 - [AWS Command Line Interfaceの設定](#)
2. AWS CLI 設定ファイルで、管理者の名前付きプロファイルを追加します。

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

AWS CLI コマンドを実行するときは、このプロファイルを使用します。名前付きプロファイルの詳細については、AWS Command Line Interface ユーザーガイドの「[名前付きプロファイル](#)」を参照してください。AWS リージョンのリストについては、<https://docs.aws.amazon.com/general/latest/gr/rande.html>の「Amazon Web Services 全般のリファレンスリージョンとエンドポイント」を参照してください。

3. コマンドプロンプトで以下のヘルプコマンドを入力して、セットアップを確認します。

```
aws help
```

次のステップ

[ステップ 3: Amazon Comprehend Medical コンソールの使用を開始する](#)

ステップ 3: Amazon Comprehend Medical コンソールの使用を開始する

Comprehend Medical コンソールの使用を開始する最も簡単な方法は、短いテキストファイルを分析することです。「[Amazon Comprehend Medical の仕組み](#)」の概念と用語を確認してから、先に進むことをお勧めします。

トピック

- [コンソールを使用した臨床テキストの分析](#)

コンソールを使用した臨床テキストの分析

Comprehend Medical コンソールでは、最大 20,000 文字の臨床テキストの内容を分析できます。分析を確認できるように、結果がコンソールに表示されます。

ドキュメントの分析を開始するには、にサインイン AWS マネジメントコンソールし、Comprehend Medical コンソールを開きます。

[Comprehend Medical] で、[Real-time analysis] (リアルタイム分析) を選択します。

コンソールに、サンプルテキストとそのテキストの分析が表示されます。

Real-time analysis [Info](#)

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

Input text

[Supported languages](#) 

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

415 of 20000 characters used.

Clear text

Analyze

サンプルテキストを英語の独自のテキストに置き換えて、[分析] を選択すると、テキストの分析を取得できます。

Insights [Info](#)

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

Analyzed text

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

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

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

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

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

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

入力テキストの下に、分析されたテキストが色分けされて、エンティティのカテゴリが示されます。

- オレンジ色のタグは PHI データを示します。

- 赤色のタグは薬剤を示します。
- 緑色のタグは病状を示します。
- 青色のタグは、検査、治療、または処置 (TTP) を示します。
- 紫色のタグは構造を示します。
- ピンク色のタグは時間表現を示します。

詳細については、「[Amazon Comprehend Medical の仕組み](#)」を参照してください。

コンソールで、入力ボックスの下にある [Analyzed Text] (分析されたテキスト) ペインに、テキストに関する詳細情報が表示されます。

[エンティティ] セクションに、テキストで検出されたエンティティのカードが表示されます。

▼ Results (27)

All ▼
< 1 2 3 4 >
⚙️

Entity ▼	Type ▼	Category ▼	Traits
87 0.9997 score	● Age	Protected health information	-
highschool teacher 0.7305 score	● Dx name	Medical condition	-
highschool teacher 0.2063 score	● Profession	Protected health information	-
<div style="border: 1px solid #ccc; padding: 2px; display: inline-block;"> □ status post 0.9525 score </div>	● Time to procedure name	Time expression	-
<div style="border: 1px solid #ccc; padding: 2px; display: inline-block;"> cardiac catheterization 0.8982 score </div>	● Procedure name	Test treatment procedure	-

各カードに、テキストとそのエンティティタイプが表示されます。

各エンティティの横にあるスコアは、表示されたエンティティのタイプとしてテキストを識別するうえで Comprehend Medical が持っている信頼度を表します。

リクエストと結果の両方の JSON 構造を表示するには、[アプリケーション統合] を選択します。JSON 構造は、オペレーションによって返される構造と同じです。

次のステップ

[ステップ 4: Amazon Comprehend Medical API の使用を開始する](#)

ステップ 4: Amazon Comprehend Medical API の使用を開始する

次の例は、Java AWS CLI、および Python を使用して Amazon Comprehend Medical オペレーションを使用する方法を示しています。これらの例を使用して、Amazon Comprehend Medical オペレーションについて独自のアプリケーションの構成要素として説明します。

AWS CLI および Python の例を実行するには、[AWS CLI](#) をインストールします。詳細については、「[ステップ 2: AWS Command Line Interface \(AWS CLI\) を設定する](#)」を参照してください。

Java の例を実行するには、AWS SDK for Java をインストールします。AWS SDK for Java をインストールする手順については、「[AWS SDK for Java のセットアップ](#)」を参照してください。

トピック

- [AWS Command Line Interface を使用した医療エンティティの検出](#)
- [AWS SDK for Java を使用した医療エンティティの検出](#)
- [AWS SDK for Python \(Boto\) を使用した医療エンティティの検出](#)

AWS Command Line Interface を使用した医療エンティティの検出

次の例は、[DetectEntitiesV2](#) オペレーションを使用して AWS CLI、テキストで検出された医療エンティティを返す方法を示しています。この例を実行するには、AWS CLI をインストールする必要があります。詳細については、「[the section called “ステップ 2: をセットアップする AWS CLI”](#)」を参照してください。

例は、Unix、Linux、および macOS 用にフォーマットされています。Windows の場合は、各行末のバックスラッシュ (\) Unix 連結文字をキャレット (^) に置き換えてください。

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

応答は以下のとおりです。

```
{
```

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

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

AWS SDK for Javaを使用した医療エンティティの検出

次の例では、Java で DetectEntitiesV2 オペレーションを使用します。この例を実行するには、AWS SDK for Javaをインストールします。のインストール手順については AWS SDK for Java、[「AWS SDK for Java のセットアップ」](#)を参照してください。

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

public class SampleAPICall {

    public static void main() {
```

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

AWSComprehendMedical client = AWSComprehendMedicalClient.builder()

.withCredentials(credentials)

.withRegion("YOUR
REGION")

.build();

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

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

出力には、入力テキスト内で検出された3つのエンティティ、および入力テキスト内のこれらのエンティティの位置が含まれます。エンティティが正しく識別された信頼レベルも、各エンティティとともにリストされます。次の出力は、前の例の Generic_Name、Dosage、Frequency の各エンティティを示しています。

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

AWS SDK for Python (Boto)を使用した医療エンティティの検出

次の例では、Python で DetectEntitiesV2 オペレーションを使用します。サンプルを実行するには、AWS CLIをインストールします。詳細については、「[the section called “ステップ 2: をセットアップする AWS CLI”](#)」を参照してください。

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

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

出力には、入力テキスト内で検出された3つのエンティティ、および入力テキスト内のこれらのエンティティの位置が含まれます。エンティティが正しく識別された信頼レベルも、各エンティティとともにリストされます。次の出力は、前の例の Generic_Name、Dosage、Frequency の各エンティティを示しています。

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

Amazon Comprehend Medical とインターフェイス VPC エンドポイント (AWS PrivateLink)

インターフェイス VPC エンドポイントを作成すると、VPC と Amazon Comprehend Medical の間にプライベート接続を確立できます。インターフェイス VPC エンドポイントは、インターネットゲートウェイ、NAT デバイス、VPN 接続、Direct Connect 接続のいずれも必要とせずに Amazon Comprehend Medical API にプライベートにアクセスするために使用できるテクノロジーである [AWS PrivateLink](#) を利用しています。VPC のインスタンスは、パブリック IP アドレスがなくても Amazon Comprehend Medical API と通信できます。VPC と Amazon Comprehend Medical API の間のトラフィックは、Amazon ネットワークを離れません。

各インターフェイスエンドポイントは、サブネット内の 1 つ以上の [Elastic Network Interface](#) によって表されます。

詳細については、「Amazon VPC ユーザーガイド」の「[インターフェイス VPC エンドポイント \(AWS PrivateLink\)](#)」を参照してください。

Amazon Comprehend Medical VPC エンドポイントに関する考慮事項

Amazon Comprehend Medical のインターフェイス VPC エンドポイントを設定する前に、Amazon VPC ユーザーガイドの「[インターフェイスエンドポイントのプロパティと制限](#)」を確認してください。

Amazon Comprehend Medical は、VPC からのすべての API アクシヨンの呼び出しをサポートしています。

Amazon Comprehend Medical のインターフェイス VPC エンドポイントの作成

Amazon Comprehend Medical サービスの VPC エンドポイントは、Amazon VPC コンソールまたは AWS Command Line Interface () を使用して作成できます AWS CLI。詳細については、Amazon VPC ユーザーガイドの [インターフェイスエンドポイントの作成](#) を参照してください。

次のサービス名を使用して、Amazon Comprehend Medical の VPC エンドポイントを作成します。

- `com.amazonaws.region.comprehendmedical`

エンドポイントのプライベート DNS を有効にすると、リージョンのデフォルト DNS 名を使用して、Amazon Comprehend Medical への API リクエストを実行できます。例えば、`comprehendmedical.us-east-1.amazonaws.com` と指定します。

詳細については、「Amazon VPC ユーザーガイド」の「[インターフェイスエンドポイントを介したサービスへのアクセス](#)」を参照してください。

Amazon Comprehend Medical の VPC エンドポイントポリシーの作成

Amazon Comprehend Medical へのアクセスを制御する VPC エンドポイントにエンドポイントポリシーをアタッチできます。このポリシーでは、以下の情報を指定します。

- アクションを実行できるプリンシパル。
- 実行可能なアクション。
- アクションを実行できるリソース。

詳細については、「Amazon VPC ユーザーガイド」の「[VPC エンドポイントでサービスへのアクセスを制御する](#)」を参照してください。

例: Amazon Comprehend Medical アクションの VPC エンドポイントポリシー

Amazon Comprehend Medical のエンドポイントポリシーの例を次に示します。このポリシーは、エンドポイントにアタッチされると、すべてのリソースのすべてのプリンシパルに対して、Amazon Comprehend Medical DetectEntitiesV2 アクションへのアクセスを許可します。

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

```
]
}
```

テキスト分析 API オペレーション

Amazon Comprehend Medical を使用すると、トレーニング済みの自然言語処理 (NLP) モデルを利用して、臨床ドキュメントを調べてコンテンツに関するさまざまなインサイトを得ることができます。分析は、単一のファイルで実行することも、Amazon Simple Storage Service (S3) バケットに保存された複数のファイルでバッチ分析として実行することもできます。

Amazon Comprehend Medical では、ドキュメントに対して次のことを実行できます。

- [エンティティの検出 \(バージョン 2\)](#) — 構造化されていない臨床テキストを調べて、病状、治療、検査と結果、薬剤などの医療情報へのテキスト参照を検出します。このバージョンでは、元の Detect entities API とは異なるモデルを使用しており、出力にもいくつかの変更があります。
- [PHI の検出](#) - 構造化されていない臨床テキストを調べて、名前や住所などの保護医療情報 (PHI) へのテキスト参照を検出します。

Amazon Comprehend Medical には、臨床文書に対してバッチテキスト分析を実行するために使用できる複数の API オペレーションも含まれています。これらの API オペレーションを使用する方法の詳細については、「[the section called “テキスト分析バッチ API”](#)」を参照してください。

トピック

- [エンティティの検出 \(バージョン 2\)](#)
- [PHI の検出](#)
- [テキスト分析バッチ API](#)

エンティティの検出 (バージョン 2)

DetectEntitiesV2 を使用して、単一ファイル内でエンティティを検出します。または、StartEntitiesDetectionV2Job を使用して、複数のファイルでバッチ分析を行います。以下のカテゴリのエンティティを検出できます。

- ANATOMY: 人体の部位または体組織への参照、およびそれらの人体の部位または体組織の位置への参照を検出します。
- BEHAVIORAL_ENVIRONMENTAL_SOCIAL: 人の健康に影響を与える環境内の行動や状態を検出します。これには、喫煙、アルコールの摂取、娯楽目的の薬物使用、アレルギー、性別、人種/民族が含まれます。

- MEDICAL_CONDITION: 病状の兆候、症状、および診断を検出します。
- MEDICATION: 患者の薬剤および投与量の情報を検出します。
- PROTECTED_HEALTH_INFORMATION: 患者の個人情報を検出します。
- TEST_TREATMENT_PROCEDURE: 病状を判断するために使用されている処置を検出します。
- TIME_EXPRESSION: 検出されたエンティティに関連付けられている時間に関連するエンティティを検出します。

6 つのすべてのカテゴリが DetectEntitiesV2 オペレーションで検出されます。PHI の検出に固有の分析では、単一ファイルで DetectPHI を使用し、バッチ分析に StartPHIDetectionJob を使用します。

Amazon Comprehend Medical は、次のクラスで情報を検出します。

- エンティティ: 人々、治療、薬剤、病状などの関連オブジェクトの名前へのテキスト参照。例えば、ibuprofen。
- カテゴリ: エンティティが属する一般化されたグループ。例えば、「Ibuprofen」(イブプロフェン) は MEDICATION カテゴリに属します。
- タイプ: 単一カテゴリ内で検出されるエンティティのタイプ。例えば、「Ibuprofen」(イブプロフェン) は MEDICATION カテゴリ内の GENERIC_NAME タイプです。
- 属性: 薬剤の投与量など、エンティティに関連する情報。例えば、200 mg は Ibuprofen (イブプロフェン) エンティティの属性です。
- 特性: Amazon Comprehend Medical がコンテキストに基づいてエンティティについて把握していること。例えば、患者が薬剤を服用していない場合、薬剤は NEGATION 特性を持ちます。
- 関係タイプ: エンティティと属性の関係。

Amazon Comprehend Medical では、入力テキスト内のエンティティの位置が表示されます。Amazon Comprehend コンソールでは、位置はグラフィカルに表示されます。API を使用する場合は、位置は数値オフセットで表示されます。

各エンティティと属性には、Amazon Comprehend Medical が検出精度に対して持っている信頼レベルを示すスコアが含まれます。各属性には関係スコアもあります。このスコアは、Amazon Comprehend Medical が属性と属性の親エンティティの関係の精度に対して持つ信頼レベルを示します。ユースケースに適した信頼しきい値を特定します。高い精度が必要な状況では、高い信頼しきい値を使用します。しきい値を満たさないデータを除外します。

人体カテゴリ

ANATOMY カテゴリは、人体の部位または体組織への参照、およびそれらの人体の部位または体組織の位置への参照を検出します。

型

- SYSTEM_ORGAN_SITE: 体組織、解剖学的位置または解剖学的部位、および人体部位。

属性

- DIRECTION: 方向性を示す用語 例えば、左、右、内側、側面、上部、下部、後部、前部、末端、近位部、対側、両側、同側、背部、腹部など。

行動、環境、社会的健康のカテゴリ

BEHAVIORAL_ENVIRONMENTAL_SOCIAL カテゴリは、人の健康に影響を与える環境内の行動や状態の言及を検出します。

タイプ

- ALCOHOL_CONSUMPTION: 患者のアルコールの摂取を使用状況、頻度、量、使用期間の観点から定義します。
- ALLERGIES: 患者のアレルギーとアレルゲンに対する反応を定義します。
- GENDER: 性同一性の特徴を示す表示。
- RACE_ETHNICITY: 患者が特定の人種や民族グループに帰属していることを示す社会政治的構成概念。
- REC_DRUG_USE: 患者の娯楽目的の薬物摂取を使用状況、頻度、量、使用期間の観点から定義します。
- TOBACCO_USE: 患者の喫煙を使用状況、頻度、量、使用期間の観点から定義します。

属性

以下の検出属性は、ALCOHOL_CONSUMPTION、TOBACCO_USE、および REC_DRUG_USE タイプにのみ適用されます。

- AMOUNT: アルコール、タバコ、娯楽用薬物の使用量。
- DURATION: アルコール、タバコ、または娯楽用薬物の使用期間。
- FREQUENCY: アルコール、タバコ、娯楽用薬物の使用頻度。

特性

以下の検出特性は、ALCOHOL_CONSUMPTION、ALLERGIES、TOBACCO_USE、および REC_DRUG_USE タイプにのみ適用されます。

- NEGATION: 結果またはアクションが否定的であるか、実行されていないことの表示。
- PAST_HISTORY: アルコール、タバコ、娯楽用薬物の使用が、患者の過去 (現在の治療を受ける前) によるものであることを示す表示。

病状カテゴリ

MEDICAL_CONDITION カテゴリは、病状の兆候、症状、および診断を検出します。このカテゴリには、1つのエンティティタイプ、4つの属性、および4つの特性があります。1つまたは複数の特性を1つのタイプに関連付けることができます。属性および属性と診断の関係に関するコンテキスト情報が検出され、RELATIONSHIP_EXTRACTION. によって DX_NAME にマッピングされます。例えば、テキスト「chronic pain in left leg」(左脚の慢性疼痛)では、「chronic」(慢性)が属性 ACUITY として検出され、「left」(左)が属性 DIRECTION として検出され、「leg」(脚)が属性 SYSTEM_ORGAN_SITE として検出されます。これらの各属性の関係は、信頼スコアとともに病状エンティティ「pain」(疼痛)にマッピングされます。

型

- DX_NAME: リストされているすべての病状。DX_NAME タイプには、現在の病気、訪問の理由、および病歴が含まれます。

属性

- ACUITY: 慢性、急性、突発性、持続性、段階的など、疾患例の決定。
- DIRECTION: 方向性を示す用語 例えば、左、右、内側、側面、上部、下部、後部、前部、末端、近位部、対側、両側、同側、背部、腹部など。
- SYSTEM_ORGAN_SITE: 解剖学的部位

- QUALITY: 病期や悪性度など、病状を説明する任意の用語。

特性

- DIAGNOSIS: 症状の原因または結果として決定される病状。症状は、身体所見、検査所見報告書や X 線診断報告書、またはその他の手段によって検出できます。
- HYPOTHETICAL: 病状が仮説として表現されているという表示。
- LOW_CONFIDENCE: 病状が不確実性が高いと表現されているという表示。これは提供された信頼度スコアとは直接関係ありません。
- NEGATION: 結果またはアクションが否定的であるか、実行されていないことの表示。
- PERTAINS_TO_FAMILY: 病状が患者ではなく患者の家族に関係しているという表示。
- SIGN: 医師が報告した病状。
- SYMPTOM: 患者が報告した病状。

薬剤カテゴリ

MEDICATION カテゴリは、患者の薬剤および投与量の情報を検出します。1 つまたは複数の属性を 1 つのタイプに適用できます。

型

- BRAND_NAME: 薬剤または治療薬の、著作権で保護されたブランド名。
- GENERIC_NAME: 薬剤または治療薬のブランド以外の名前、成分名、または処方調合。

属性

- DOSAGE: 指示された薬剤の量。
- DURATION: 薬剤を投与すべき期間。
- FORM: 薬剤の形態。
- FREQUENCY: 薬剤を投与する頻度。
- RATE: 薬剤の投与速度 (主に薬剤注入または IV の場合)
- ROUTE_OR_MODE: 薬剤の投与方法。
- STRENGTH: 薬剤の強度。

特性

- NEGATION: 患者が薬剤を服用していないことを示す表示。
- PAST_HISTORY: 検出された薬剤が患者の過去 (現在の治療を受ける前) のものであることを示す表示。

保護医療情報カテゴリ

PROTECTED_HEALTH_INFORMATION カテゴリは、患者の個人情報を検出します。このオペレーションの詳細については、「[PHI の検出](#)」を参照してください。

型

- ADDRESS: 施設、部門、または施設内の病棟の住所のすべての地域区分。
- AGE: 記載された年齢のすべての構成要素、年齢層、または年齢。これには、患者、家族、またはそれ以外の人のものが含まれます。特に明記されていない限り、デフォルトは年単位です。
- EMAIL: E メールアドレス。
- ID: 社会保障番号、医療記録番号、施設識別番号、臨床試験番号、証明書番号や免許証番号、車両番号や機器番号、ケアの場所、または医療提供者。これには、身長、体重、検査値など、患者の生体認証番号も含まれます。
- NAME: すべての名前。通常、患者、家族、または医療提供者の名前。
- PHONE_OR_FAX: 電話番号、FAX 番号、またはポケットベル番号。1-800-QUIT-NOW や 911 など、名前付きの電話番号は除きます。
- PROFESSION: 患者または患者の家族に関連する職業または雇用主。これには、臨床記録に記載された臨床医の職業は含まれません。

検査、治療、処置のカテゴリ

TEST_TREATMENT_PROCEDURE カテゴリは、病状を判断するために使用されている処置を検出します。1 つまたは複数の属性を TEST_NAME タイプの 1 つのエンティティに関連付けることができます。

型

- PROCEDURE_NAME: 病状を治療するために、または患者ケアを提供するために、患者に行われる 1 回限りのアクションとしての治療介入。

- TEST_NAME: 値が得られる可能性がある、診断、測定、スクリーニング、または評価のために患者に行われる処置。これには、診断の決定、病状の除外や検出、または患者のスケールやスコアリングのための処置、プロセス、評価、または評価が含まれます。
- TREATMENT_NAME: 病気や障害に対処するために長期間にわたって行われる治療介入。これには、抗ウイルス薬や予防接種などの薬剤のグループが含まれます。

属性

- TEST_VALUE: 検査結果。TEST_NAME エンティティタイプにのみ適用されます。
- TEST_UNIT: 検査の値に付随する可能性のある測定単位。TEST_NAME エンティティタイプにのみ適用されます。

特性

- FUTURE: 検査、治療、または処置が、メモの対象の後に行われる行為または事象を指すことを示す表示。
- HYPOTHETICAL: 検査、治療、または処置が仮説として表現されていることを示す表示。
- NEGATION: 結果またはアクションが否定的であるか、実行されていないことの表示。
- PAST_HISTORY: 検査、治療、または処置が患者の過去（現在の治療を受ける前）のものであることを示す表示。

時間表現のカテゴリ

TIME_EXPRESSION カテゴリは、時間に関連するエンティティを検出します。これには、「3日前」、「今日」、「現在」、「入院日」、「先月」、「16日間」など、日付や時間表現などのエンティティが含まれます。このカテゴリの結果は、エンティティに関連付けられている場合にのみ返されます。例えば、「Yesterday, the patient took 200 mg of ibuprofen」(昨日、患者は 200mg のイブプロフェンを服用した) は、Yesterday を TIME_EXPRESSION エンティティとして返し、このエンティティは GENERIC_NAME エンティティ「ibuprofen」(イブプロフェン) と重複します。ただし、「yesterday, the patient walked their dog.」(昨日、患者は犬を散歩させた) では、これはエンティティとして認識されません。

型

- TIME_TO_MEDICATION_NAME: 薬剤を服用した日付。このタイプに固有の属性は BRAND_NAME および GENERIC_NAME です。
- TIME_TO_DX_NAME: 病状が発生した日付。このタイプの属性は DX_NAME です。
- TIME_TO_TEST_NAME: 検査が行われた日付。このタイプの属性は TEST_NAME です。
- TIME_TO_PROCEDURE_NAME: 措置が行われた日付。このタイプの属性は PROCEDURE_NAME です。
- TIME_TO_TREATMENT_NAME: 治療が行われた日付。このタイプの属性は TREATMENT_NAME です。

関係タイプ

- エンティティと属性の関係。認識される Relationship_type は、次のとおりです。
Overlap — TIME_EXPRESSION が検出されたエンティティと一致します。

PHI の検出

DetectPHI オペレーションは、臨床テキストのスキャン時に保護医療情報 (PHI) データを検出する場合にのみ使用します。臨床テキスト内の利用可能なエンティティをすべて検出するには、DetectEntitiesV2 を使用してください。

この API は PHI エンティティの検出のみが必要なユースケースに最適です。PHI 以外のカテゴリの詳細については、「[エンティティの検出 \(バージョン 2\)](#)」を参照してください。

Important

Amazon Comprehend Medical は、検出されたエンティティの精度に対する信頼度を示す信頼スコアを提供します。これらの信頼スコアを評価し、ユースケースに適した信頼しきい値を特定します。特定のコンプライアンスのユースケースについては、追加で人間によるレビューまたはその他の手法を使用して、検出された PHI の精度を確認することをお勧めします。

HIPAA 法では、18 個の識別子のリストに基づいた PHI を特に注意して扱う必要があります。Amazon Comprehend Medical は、これらの識別子に関連付けられたエンティティを検出しま

すが、これらのエンティティはセーフハーバー方式で指定されているリストに 1 対 1 でマッピングされません。すべての識別子が構造化されていない臨床テキストに含まれるわけではありませんが、Amazon Comprehend Medical は関連するすべての識別子に対応しています。これらの識別子は、以下のリストを含め、個々の患者を特定するために使用できるデータで構成されています。詳細については、米国政府の Health and Human Services ウェブサイトの [Health Information Privacy](#) を参照してください。

PHI 関連の各エンティティには、Amazon Comprehend Medical が検出精度に対して持っている信頼レベルを示すスコア (レスポンス内の Score) が含まれます。ユースケースに適した信頼しきい値を特定し、それを満たしていないエンティティを除外します。PHI の存在を特定するときは、フィルタリングに低い信頼しきい値を使用して、より多くの数の検出されたエンティティをキャプチャすることをお勧めします。これは、コンプライアンスのユースケースで検出されたエンティティの値を使用しない場合に、特に当てはまります。

以下の PHI 関連エンティティは、DetectPHI オペレーションまたは DetectEntitiesV2 オペレーションを実行することにより検出できます。

検出される PHI エンティティ

エンティティ	説明	HIPAA カテゴリ
AGE	患者、家族、または記録に関係しているそれ以外の人について、記載された年齢のすべての構成要素、年齢層、および年齢。特に明記されていない限り、デフォルトは年単位です。	3. 個人に関連する日付
DATE	患者または患者ケアに関連する日付。	3. 個人に関連する日付
NAME	臨床ノートに記載されているすべての名前。通常、患者、家族、または医療提供者に属する名前。	1. 名前
PHONE_OR_FAX	電話、ファックス、ポケットベル。1-800-QUIT-NOW や	4. 電話番号 5. ファックス番号

エンティティ	説明	HIPAA カテゴリ
	911 などの名前付き電話番号は除きます。	
EMAIL	E メールアドレス。	6. E メールアドレス
ID	患者の身元に関連付けられた、あらゆる種類の番号。これには、社会保障番号、医療記録番号、施設識別番号、臨床試験番号、証明書番号や免許証番号、車両番号や機器番号が含まれます。また、生体認証番号、およびケアの場所や医療提供者を識別する番号も含まれます。	7. 社会保障番号 8. 医療記録番号 9. 健康保険番号 10. 口座番号 11. 証明書/免許証番号 12. 車両識別番号 13. 機器番号 16. バイオメトリック情報 18. その他の識別特性
[URL]	任意のウェブ URL。	14. URL
ADDRESS	これには、施設、名前付き医療施設、または施設内の病棟の住所のすべての地域区分が含まれます。	2. 地理的場所
PROFESSION	患者または患者の家族に関連する、臨床記録に記載された職業または雇用主が含まれます。	18. その他の識別特性

例

テキスト「Patient is John Smith, a 48-year-old teacher and resident of Seattle, Washington.」(患者は 48 歳の教師でワシントン州シアトルの居住者であるジョン・ スミスです) 上のリクエストは次のオブジェクトを返します。

- 「John Smith」(ジョン・ スミス) を PROTECTED_HEALTH_INFORMATION カテゴリのタイプ NAME のエンティティとして返します。
- 「48」を PROTECTED_HEALTH_INFORMATION カテゴリのタイプ AGE のエンティティとして返します。
- 「teacher」(教師) を PROTECTED_HEALTH_INFORMATION カテゴリのタイプ PROFESSION (識別特性) のエンティティとして返します。
- 「Seattle, Washington」(ワシントン州シアトル) を PROTECTED_HEALTH_INFORMATION カテゴリの ADDRESS エンティティとして返します。

Amazon Comprehend Medical コンソールでは、次のように表示されます。

```
Patient is John Smith, a 48 year old teacher and resident  
• Name (John Smith) • Age (48) • Profession (teacher)  
  
of Seattle, Washington.  
• Address (Seattle , Washington)
```

DetectPHI オペレーションを使用する場合、レスポンスは次のようになります。StartPHIDetectionJob オペレーションを使用する場合、Amazon Comprehend Medical はこの構造で出力場所にファイルを作成します。

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

```
    "EndOffset": 27,
    "Score": 0.9998362064361572,
    "Text": "48",
    "Category": "PROTECTED_HEALTH_INFORMATION",
    "Type": "AGE",
    "Traits": []
  },
  {
    "Id": 2,
    "BeginOffset": 37,
    "EndOffset": 44,
    "Score": 0.8661606311798096,
    "Text": "teacher",
    "Category": "PROTECTED_HEALTH_INFORMATION",
    "Type": "PROFESSION",
    "Traits": []
  },
  {
    "Id": 3,
    "BeginOffset": 61,
    "EndOffset": 68,
    "Score": 0.9629441499710083,
    "Text": "Seattle",
    "Category": "PROTECTED_HEALTH_INFORMATION",
    "Type": "ADDRESS",
    "Traits": []
  },
  {
    "Id": 4,
    "BeginOffset": 78,
    "EndOffset": 88,
    "Score": 0.38217034935951233,
    "Text": "Washington",
    "Category": "PROTECTED_HEALTH_INFORMATION",
    "Type": "ADDRESS",
    "Traits": []
  }
],
"UnmappedAttributes": []
}
```

テキスト分析バッチ API

Amazon Comprehend Medical を使用して、Amazon S3 バケットに保存されている医療テキストを分析します。1つのバッチで最大 10 GB のドキュメントを分析します。コンソールを使用してバッチ分析ジョブを作成および管理するか、またはバッチ API を使用して保護医療情報 (PHI) などの医療エンティティを検出します。API は、継続中のバッチ分析ジョブの開始、停止、一覧表示、および説明を実行します。

バッチ分析およびその他の Amazon Comprehend Medical オペレーションの価格情報については、[こちら](#)を参照してください。

重要な注意点

Amazon Comprehend Medical のバッチ分析オペレーションは、専門家による医療の助言、診断、または治療の代用品ではありません。ユースケースに適した信頼しきい値を特定し、高い精度を必要とする状況では高い信頼しきい値を使用してください。特定のユースケースでは、適切な訓練を受けたレビュー担当者によって人的に結果を見直し、検証する必要があります。Amazon Comprehend Medical のすべてのオペレーションは、訓練を受けた医療専門家による正確さと健全な医療判断の確認後、患者ケアのシナリオでのみ使用してください。

API を使用したバッチ分析の実行

バッチ分析ジョブは、Amazon Comprehend Medical コンソールまたは Amazon Comprehend Medical バッチ API を使用して実行できます。

前提条件

Amazon Comprehend Medical API を使用する場合、AWS Identity Access and Management (IAM) ポリシーを作成し、このポリシーを IAM ロールにアタッチします。IAM ロールと信頼ポリシーの詳細については、「[IAM でのポリシーとアクセス許可](#)」を参照してください。

1. データを S3 バケットにアップロードします。
2. 新しい分析ジョブを開始するには、StartEntitiesDetectionV2Job オペレーションまたは StartPHIDetectionJob オペレーションを使用します。ジョブを開始するときに、入力ファイルが含まれた入力 S3 バケットの名前を Amazon Comprehend Medical に指定し、バッチ分析後にファイルを書き込む出力 S3 バケットを指定します。
3. コンソールを使用するか、DescribeEntitiesDetectionV2Job オペレーションまたは DescribePHIDetectionJob オペレーションを使用して、ジョブの進行状況を監視します。ま

た、ListEntitiesDetectionV2Jobs および ListPHIDetectionJobs を使用して、オントロジーリンクのすべてのバッチ分析ジョブのステータスを確認できます。

4. 進行中のジョブを停止する必要がある場合は、StopEntitiesDetectionV2Job または StopPHIDetectionJob を使用して分析を停止します。
5. 分析ジョブの結果を表示するには、ジョブの開始時に設定した出力 S3 バケットを確認します。

コンソールを使用したバッチ分析の実行

1. データを S3 バケットにアップロードします。
2. 新しい分析ジョブを開始するには、実行する分析のタイプを選択します。入力ファイルが含まれる S3 バケットの名前と、出力ファイルを送信する S3 バケットの名前を指定します。
3. 進行中のジョブのステータスを監視します。コンソールから、分析の開始日と終了日など、すべてのバッチ分析オペレーションとそのステータスを表示できます。
4. 分析ジョブの結果を表示するには、ジョブの開始時に設定した出力 S3 バケットを確認します。

バッチオペレーションの IAM ポリシー

Amazon Comprehend Medical バッチ API を呼び出す IAM ロールには、入力ファイルと出力ファイルが含まれる S3 バケットへのアクセス権を付与するポリシーが必要です。Amazon Comprehend Medical サービスがロールを引き受けることができるようにする信頼関係も割り当てる必要があります。IAM ロールと信頼ポリシーの詳細については、「[IAM ロール](#)」を参照してください。

ロールには、以下のポリシーが必要です。

JSON

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

```

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

```

ロールには、以下の信頼関係が必要です。混乱した代理によるセキュリティの問題を防ぐために `aws:SourceAccount` および `aws:SourceArn` 条件キーを使用することをお勧めします。混乱した代理問題と AWS アカウントを保護する方法の詳細については、IAM [ドキュメントの混乱した代理問題](#)を参照してください。

JSON

```

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

```

```
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "account_id"
      },
      "ArnLike": {
        "aws:SourceArn": "arn:aws:comprehendmedical:us-
east-1:account_id:*"
      }
    }
  ]
}
```

バッチ分析の出カファイル

Amazon Comprehend Medical は、バッチ内の入力ファイルごとに 1 つの出カファイルを作成します。ファイルの拡張子は .out です。Amazon Comprehend Medical は、まず、*AwsAccountId-JobType-JobId* を名前にして出力 S3 バケット内にディレクトリを作成し、バッチのすべての出カファイルをこのディレクトリに書き込みます。Amazon Comprehend Medical は、ジョブの出力が別のジョブの出力を上書きしないように、この新しいディレクトリを作成します。

バッチオペレーションの出力では、同期オペレーションと同じ出力が生成されます。Amazon Comprehend Medical によって生成される出力の例については、「[エンティティの検出 \(バージョン 2\)](#)」を参照してください。

バッチオペレーションごとに、ジョブに関する情報が含まれた 3 つのマニフェストファイルが生成されます。

- Manifest — ジョブを要約します。ジョブに使用されたパラメータ、ジョブの合計サイズ、および処理されたファイル数に関する情報を提供します。
- success — 処理が成功したファイルに関する情報を提供します。入力ファイル名と出力ファイル名、および入力ファイルのサイズが含まれます。
- unprocessed — バッチジョブで処理されなかったファイルをリスト表示します。ファイルごとのエラーコードとエラーメッセージも含まれます。

Amazon Comprehend Medical は、バッチジョブ用に指定された出力ディレクトリにこれらのファイルを書き込みます。概要マニフェストファイルは、Manifest_AccountId-Operation-JobId というタイトルのフォルダと共に出力フォルダに書き込まれます。マニフェストフォルダ内には、成功マニフェストを含む success フォルダがあります。また、未処理のファイルマニフェストを含む failed フォルダも含まれます。以降のセクションでは、マニフェストファイルの構造を示します。

バッチのマニフェストファイル

以下に、バッチのマニフェストファイルの JSON 構造を示します。

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

成功のマニフェストファイル

以下に、処理が成功したファイルに関する情報が含まれたファイルの JSON 構造を示します。

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

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

未処理のマニフェストファイル

以下に、未処理のファイルに関する情報が含まれたマニフェストファイルの JSON 構造を示します。

```
{
  "Files" : [ {
    "Input": "file_name_that_failed",
    "ErrorCode": "error code for exception",
    "ErrorMessage": "explanation of the error code and suggestions"
  },
  { ...}
]
}
```

オントロジーリンク

Amazon Comprehend Medical では、臨床テキスト内のエンティティを検出し、これらのエンティティを RxNorm、ICD-10-CM、および SNOMED CT のナレッジベースなどの標準化された医療オントロジー内の概念にリンクできます。分析は、単一のファイルで実行することも、Amazon Simple Storage Service (S3) に保存された大きいドキュメントや複数のファイルでバッチ分析として実行することもできます。

ICD-10-CM のリンク

InferICD10CM を使用して、可能性のある病状をエンティティとして検出し、2026 年版の [国際疾病分類第 10 回リビジョン](#)、[臨床変更 \(ICD-10-CM\)](#) のコードにリンクします。ICD-10-CM は米国疾病管理予防センター (CDC) によって提供されています。

病状が検出された場合、InferICD10CM は病状が一致する ICD-10-CM コードと説明を返します。検出された病状は、信頼度の高い順でリストされます。スコアは、概念に一致したエンティティの精度に対する信頼度を示します。徴候、症状、否定などの関連情報は、特性として認識されます。解剖指定や重度などの追加情報は、属性としてリストされます。

InferICD10CM は、以下のシナリオに最適です。

- 患者記録のための専門的な医療事務の支援
- 臨床研究と臨床試験
- 医療ソフトウェアシステムとの統合
- 早期発見と診断
- 公衆衛生管理

ICD-10-CM のカテゴリ

InferICD10CM は、MEDICAL_CONDITION カテゴリのエンティティを検出します。追加の関連情報も検出され、属性または特性としてリンクされます。

ICD-10-CM のタイプ

InferICD10CM は、DX_NAME タイプおよび TIME_EXPRESSION タイプのエンティティを検出します。

ICD-10-CM の特性

InferICD10CM は、以下のコンテキスト情報を特性として検出します。

- DIAGNOSIS: 症状の評価によって決定される病状の識別。
- HYPOTHETICAL: 病状が仮説として表現されているという表示。
- LOW_CONFIDENCE: 病状が不確実性が高いと表現されているという表示。これは提供された信頼度スコアとは直接関係ありません。
- NEGATION: 病状がないという表示。
- PERTAINS_TO_FAMILY: 病状が患者ではなく患者の家族に関係しているという表示。
- SIGN: 医師が報告した病状。
- SYMPTOM: 患者が報告した病状。

ICD-10-CM の属性

InferICD10CM は、以下のコンテキスト情報を属性として検出します。

- DIRECTION: 方向性を示す用語 例えば、左、右、内側、側面、上部、下部、後部、前部、末端、近位部、対側、両側、同側、背部、腹部など。
- SYSTEM_ORGAN_SITE: 解剖学的部位
- ACUITY: 慢性、急性、突発性、持続性、段階的など、疾患例の決定。MEDICAL_CONDITION タイプにのみ適用されます。
- QUALITY: 病期や悪性度など、病状を説明する任意の用語。

時間表現のカテゴリ

TIME_EXPRESSION カテゴリは、時間に関連するエンティティを検出します。これには、「3日前」、「今日」、「現在」、「入院日」、「先月」、「16日間」など、日付や時間表現などのエンティティが含まれます。このカテゴリの結果は、エンティティに関連付けられている場合にのみ返されます。例えば、「Yesterday, the patient was diagnosed with influenza」(昨日、患者がインフルエンザと診断された) という表現では、Yesterday が TIME_EXPRESSION エンティティとして返され、このエンティティは DX_NAME エンティティ「influenza」(インフルエンザ) と重複しています。ただし、「yesterday, the patient walked their dog.」(昨日、患者は犬を散歩させた) という表現では、「yesterday」(昨日) はエンティティとして認識されません。

型

認識されるタイプの TIME_EXPRESSION は、病状が発生した日である TIME_TO_DX_NAME です。このタイプの属性は DX_NAME です。

関係タイプ

RELATIONSHIP_TYPE は、エンティティと属性の関係を指します。認識される RELATIONSHIP_TYPE は OVERLAP であり、TIME_EXPRESSION は検出されたエンティティと一致します。

入力とレスポンスの例

Note

特定の API 入力および応答構文については、「Amazon Comprehend Medical API リファレンス」の「[InferICD10CM](#)」を参照してください。

以下の入力テキストの例は、InferICD10CM オペレーションがどのように機能するかを示しています。すべての入力テキストを表示するには、[コピー] ボタンにスクロールします。

```
"The patient is a 71-year-old female patient of Dr. X. The patient presented to the emergency room last evening with approximately 7 to 8 day history of abdominal pain which has been persistent. She has had no nausea and vomiting, but has had persistent associated anorexia. She is passing flatus, but had some obstipation symptoms with the last bowel movement two days ago. She denies any bright red blood per rectum and no history of recent melena. Her last colonoscopy was approximately 5 years ago with Dr. Y. She has had no definite fevers or chills and no history of jaundice. The patient denies any significant recent weight loss."
```

InferICD10CM オペレーションは JSON 形式で以下の出力を返します (簡潔にするために省略しています)。

```
{
  "Entities": [
    {
      "Id": 1,
      "Text": "abdominal pain",
      "Category": "MEDICAL_CONDITION",
      "Type": "DX_NAME",
```

```
"Score": Float,
"BeginOffset": 153,
"EndOffset": 167,
"Attributes": [
  {
    "Type": "ACUITY",
    "Score": Float,
    "RelationshipScore": Float,
    "Id": 2,
    "BeginOffset": 183,
    "EndOffset": 193,
    "Text": "persistent",
    "Traits": []
  }
],
"Traits": [
  {
    "Name": "SYMPTOM",
    "Score": Float
  }
],
"ICD10CMConcepts": [
  {
    "Description": "Unspecified abdominal pain",
    "Code": "R10.9",
    "Score": Float
  },
  {
    "Description": "Epigastric pain",
    "Code": "R10.13",
    "Score": Float
  },
  {
    "Description": "Lower abdominal pain, unspecified",
    "Code": "R10.30",
    "Score": Float
  },
  {
    "Description": "Generalized abdominal pain",
    "Code": "R10.84",
    "Score": Float
  },
  {
    "Description": "Upper abdominal pain, unspecified",
```

```
        "Code": "R10.10",
        "Score": Float
    }
]
}
...
"ModelVersion": "3.2.0.20251001"
}
```

また、InferICD10CM は、エンティティがテキスト内で否定されている場合も認識します。例えば、患者に症状がない場合、症状と否定の両方が特性として識別されて、信頼スコアとともにリストされます。前の例の入力に基づいて、患者には nausea (吐き気) がないため、症状 Nausea が NEGATION の下にリストされます。

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

```
    }  
  ]  
}
```

RxNorm のリンク

InferRxNorm オペレーションを使用して、患者記録にリストされている薬剤をエンティティとして識別します。また、このオペレーションは、これらのエンティティを[米国国立医学図書館の RxNorm データベース](#)の概念識別子 (RxCUI) にリンクします。各 RxCUI のソースは 2022-11-07 RxNorm および RxTerms リリースです。各 RxCUI は、さまざまな強度と処方に対して一意です。Amazon Comprehend Medical は、信頼スコアに基づいて、検出された各薬剤について一致する可能性のある上位の RxCUI を降順にリストします。RxCUI コードは、構造化されていないテキストでは不可能なダウンストリーム分析に使用します。強度、頻度、投与量、処方、投与経路などの関連情報は、JSON 形式で属性としてリストされます。

以下のシナリオで InferRxNorm を使用できます。

- 患者が服用した薬剤のスクリーニング。
- 新たに処方された薬剤と患者が現在服用している薬剤による、潜在的な負の反応の防止。
- RxCUI を用いた薬歴に基づいた、臨床試験に含めるかどうかのスクリーニング。
- 薬剤の投与量と頻度が適切かどうかの確認。
- 薬剤の使用、効能、および副作用のスクリーニング。
- 公衆衛生の管理。

重要な注意点

Amazon Comprehend Medical の InferRxNorm オペレーションは、専門家による医療の助言、診断、または治療の代用品ではありません。ユースケースに適した信頼しきい値を特定し、高い精度を必要とする状況では高い信頼しきい値を使用してください。Amazon Comprehend Medical オペレーションは、訓練を受けた医療専門家による結果の正確さと健全な判断の確認後に、患者ケアのシナリオでのみ使用してください。

RxNorm のカテゴリ

InferRxNorm は、MEDICATION カテゴリのエンティティを検出します。属性または特性としてリンクされる追加の関連情報も検出されます。

RxNorm のタイプ

Medication カテゴリのエンティティのタイプは、次のとおりです。

- BRAND_NAME: 薬剤または治療薬の、著作権で保護されたブランド名。
- GENERIC_NAME: 薬剤または治療薬のブランド以外の名前、成分名、または処方調合。

RxNorm の属性

- DOSAGE: 指示された薬剤の量。
- DURATION: 薬剤を投与すべき期間。
- FORM: 薬剤の形態。
- FREQUENCY: 薬剤を投与する頻度。
- RATE: 薬剤の投与速度 (主に薬剤注入または IV の場合)
- ROUTE_OR_MODE: 薬剤の投与方法。
- STRENGTH: 薬剤の強度。

RxNorm の特性

- NEGATION: 患者が薬剤を服用していないことを示す表示。
- PAST_HISTORY: 検出された薬剤が患者の過去 (現在の治療を受ける前) のものであることを示す表示。

入力とレスポンスの例

Note

特定の API 入力および応答構文については、「Amazon Comprehend Medical API リファレンス」の「[InferRxNorm](#)」を参照してください。

以下の入力テキストの例は、InferRxNorm オペレーションがどのように機能するかを示しています。すべての入力テキストを表示するには、[コピー] ボタンにスクロールします。

```
"fluoride topical ( fluoride 1.1 % topical gel ) 1 application Topically daily Brush
onto teeth before bed time , spit , do not rinse, eat or drink for 20-30 minutes"
```

InferRxNorm オペレーションにより、JSON 形式で次の出力が返されます。

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

```
    ]
  }
],
"ModelVersion": "3.2.0.20221107"
}
```

InferRxNorm オペレーションは次の入力テキストを使用して、否定の特性も認識します。

```
'patient is not on warfarin'
```

InferRxNorm オペレーションにより、JSON 形式で次の出力が返されます。

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

```
        "Score": Float
    },
    {
        "Description": "warfarin sodium 2 MG Oral Tablet [Coumadin]",
        "Code": "855304",
        "Score": Float
    },
    {
        "Description": "warfarin sodium 10 MG Oral Tablet [Jantoven]",
        "Code": "855300",
        "Score": Float
    }
]
}
],
"ModelVersion": "3.2.0.20221107"
}
```

SNOMED CT のリンク

InferSNOMEDCT を使用して、医療概念を識別し、2022-03 バージョンの Systematized Nomenclature of Medicine, Clinical Terms (SNOMED CT) のコードにリンクします。SNOMED CT は、病状や解剖学、医学的検査、治療、処置など、医療概念の包括的な語彙を提供します。SNOMED CT の詳細については、[SNOMED CT](#) のサイトを参照してください。

Amazon Comprehend Medical では、検出された各医療エンティティについて、その医療概念に関連する上位 5 つの SNOMED CT 概念 ID と説明、およびその予測におけるモデルの信頼性を示す信頼度スコアがリストされます。SNOMED CT 概念 ID は、信頼スコアと共に信頼度の高い順でリストされます。SNOMED CT の概念 ID を SNOMED CT 多重階層と併用すると、医療事務、レポート、または臨床分析用に患者の臨床データの構造化できます。

InferSNOMEDCT は米国のお客様にご利用いただけます。他の国における SNOMED CT に関する情報、および SNOMED CT のライセンスに関する情報については、[SNOMED CT](#) のサイトを参照してください。

InferSNOMEDCT は、以下のシナリオに最適です。

- 患者記録のための専門的な医療事務の支援
- 臨床研究と臨床試験
- 公衆衛生管理

InferSNOMEDCT によって、以下のカテゴリのエンティティが検出されます。追加の関連情報も検出され、属性または特性としてリンクされます。

- MEDICAL_CONDITION: 病状の兆候、症状、および診断。
- ANATOMY: 人体の部位または体組織、およびそれらの人体の部位または体組織の位置。
- TEST_TREATMENT_PROCEDURE: 病状を判断するために使用されている処置。

人体カテゴリ

ANATOMY カテゴリは、人体の部位または体組織への参照、およびそれらの人体の部位または体組織の位置への参照を検出します。

属性

ANATOMY カテゴリでは以下の属性が検出されます。

- DIRECTION: 方向性を示す用語 例えば、左、右、内側、側面、上部、下部、後部、前部、末端、近位部、対側、両側、同側、背部、腹部など。
- SYSTEM_ORGAN_SITE: 体組織、解剖学的位置または解剖学的部位、および人体部位。

病状カテゴリ

MEDICAL_CONDITION カテゴリは、病状の兆候、症状、および診断を検出します。

タイプ

MEDICAL_CONDITION カテゴリでは、次のタイプが検出されます。

- DX_NAME: 症状の評価によって決定される病状の識別。

属性

MEDICAL_CONDITION カテゴリでは以下の属性が検出されます。

- ACUITY: 慢性、急性、突発性、持続性、段階的など、疾患例の決定。
- QUALITY: 病期や悪性度など、病状を説明する任意の用語。
- DIRECTION: 方向性を示す用語 例えば、左、右内側、側面、上部、下部、後部、前部、末端、近位部、対側、両側、同側、背部、腹部など。

- SYSTEM_ORGAN_SITE: 体組織、解剖学的位置または解剖学的部位、および人体部位。

特性

MEDICAL_CONDITION カテゴリでは以下の特性が検出されます。

- DIAGNOSIS: 症状の原因または結果として決定される病状。症状は、身体所見、検査所見報告書や X 線診断報告書、またはその他の手段によって検出できます。
- HYPOTHETICAL: 病状が仮説として表現されているという表示。
- LOW_CONFIDENCE: 病状が不確実性が高いと表現されているという表示。これは提供された信頼度スコアとは直接関係ありません。
- NEGATION: 病状がないという表示。
- PERTAINS_TO_FAMILY: 病状が患者ではなく患者の家族に関係しているという表示。
- SIGN: 医師が報告した病状。
- SYMPTON: 患者が報告した病状。

検査、治療、処置のカテゴリ

TEST_TREATMENT_PROCEDURE カテゴリは、病状を判断するために使用されている処置を検出します。

タイプ

TEST_TREATMENT_PROCEDURE カテゴリでは、以下のタイプが検出されます。

- PROCEDURE_NAME: 病状を治療するために、または患者ケアを提供するために、患者に対して行われる治療介入。
- TEST_NAME: 値が得られる可能性がある、診断、測定、スクリーニング、または評価のために患者に行われる処置。これには、診断の決定、病状の除外や検出、または患者のスケールやスコアリングのための処置、プロセス、評価、または評価が含まれます。
- TREATMENT_NAME: 病気や障害に対処するために行われる治療介入。これには、抗ウイルス薬や予防接種などの薬剤が含まれます。

属性

TEST_TREATMENT_PROCEDURE カテゴリでは、以下の属性が検出されます。

- TEST_NAME: 診断検査が実行されました。
- TEST_VALUE: 診断検査の結果の数値。
- TEST_UNIT: 結果の TEST_VALUE: に関連する単位。
- PROCEDURE_NAME: 実施された手術または医療処置の名称。
- TREATMENT_NAME: 患者に施される治療法の名称。

特性

- FUTURE: 検査、治療、または処置が、メモの対象の後に行われる行為または事象を指すことを示す表示。
- HYPOTHETICAL: 検査、治療、または処置が仮説として表現されていることを示す表示。
- NEGATION: 結果またはアクションが否定的であるか、実行されていないことの表示。
- PAST_HISTORY: 検査、治療、または処置が患者の過去（現在の治療を受ける前）のものであることを示す表示。

SNOMED CT の詳細

JSON 応答には、以下の情報を含む SNOMED CT の詳細が含まれます。

- EDITION: US エディションのみがサポートされます。
- VERSIONDATE: 使用した SNOMED CT バージョンの日付スタンプ。
- LANGUAGE: 英語 (US-EN) 言語での分析がサポートされています。

入力とレスポンスの例

Note

特定の API 入力および応答構文については、「Amazon Comprehend Medical API リファレンス」の「[InferSNOMEDCT](#)」を参照してください。

以下の入力テキストの例は、InferSNOMEDCT オペレーションがどのように機能するかを示しています。すべての入力テキストを表示するには、[コピー] ボタンにスクロールします。

```
"HEENT : Boggy inferior turbinates, No oropharyngeal lesion"
```

InferSNOMEDCT オペレーションにより、JSON 形式で次の出力が返されます。

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

```
    "Id": 0
  },
  {
    "Category": "MEDICAL_CONDITION",
    "BeginOffset": 8,
    "EndOffset": 33,
    "Text": "Boggy inferior turbinates",
    "Traits": [
      {
        "Score": Float,
        "Name": "SIGN"
      }
    ],
    "SNOMEDCTConcepts": [
      {
        "Code": "254477009",
        "Score": Float,
        "Description": "Tumor of inferior turbinate (disorder)"
      },
      {
        "Code": "260762006",
        "Score": Float,
        "Description": "Choroidal invasion status (attribute)"
      },
      {
        "Code": "2455009",
        "Score": Float,
        "Description": "Revision of lumbosubarachnoid shunt (procedure)"
      },
      {
        "Code": "19883003",
        "Score": Float,
        "Description": "Atrophy of nasal turbinates (disorder)"
      },
      {
        "Code": "256723009",
        "Score": Float,
        "Description": "Inferior turbinate flap (substance)"
      }
    ],
    "Score": Float,
    "Attributes": [
      {
        "Category": "ANATOMY",
```

```

    "RelationshipScore": Float,
    "EndOffset": 5,
    "Text": "HEENT",
    "Traits": [],
    "SNOMEDCTConcepts": [
      {
        "Code": "69536005",
        "Score": Float,
        "Description": "Head structure (body structure)"
      },
      {
        "Code": "429031000124106",
        "Score": Float,
        "Description": "Review of systems, head, ear, eyes, nose
and throat (procedure)"
      },
      {
        "Code": "385383008",
        "Score": Float,
        "Description": "Ear, nose and throat structure (body
structure)"
      },
      {
        "Code": "64237003",
        "Score": Float,
        "Description": "Structure of left half of head (body
structure)"
      },
      {
        "Code": "113028003",
        "Score": Float,
        "Description": "Ear, nose and throat examination
(procedure)"
      }
    ],
    "Score": Float,
    "RelationshipType": "SYSTEM_ORGAN_SITE",
    "Type": "SYSTEM_ORGAN_SITE",
    "Id": 0,
    "BeginOffset": 0
  }
],
  "Type": "DX_NAME",
  "Id": 1

```

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

```

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

```

```
    }
  ],
  "SNOMEDCTConcepts": [
    {
      "Code": "31389004",
      "Score": Float,
      "Description": "Oropharyngeal structure (body structure)"
    },
    {
      "Code": "33431000119109",
      "Score": Float,
      "Description": "Lesion of oropharynx (disorder)"
    },
    {
      "Code": "764786007",
      "Score": Float,
      "Description": "Oropharyngeal (intended site)"
    },
    {
      "Code": "418664002",
      "Score": Float,
      "Description": "Oropharyngeal route (qualifier value)"
    },
    {
      "Code": "110162001",
      "Score": Float,
      "Description": "Abrasion of oropharynx (disorder)"
    }
  ],
  "Score": Float,
  "Attributes": [
    {
      "Category": "ANATOMY",
      "RelationshipScore": Float,
      "EndOffset": 5,
      "Text": "HEENT",
      "Traits": [],
      "SNOMEDCTConcepts": [
        {
          "Code": "69536005",
          "Score": Float,
          "Description": "Head structure (body structure)"
        },
        {
```

```

        "Code": "429031000124106",
        "Score": Float,
        "Description": "Review of systems, head, ear, eyes, nose
and throat (procedure)"
    },
    {
        "Code": "385383008",
        "Score": Float,
        "Description": "Ear, nose and throat structure (body
structure)"
    },
    {
        "Code": "64237003",
        "Score": Float,
        "Description": "Structure of left half of head (body
structure)"
    },
    {
        "Code": "113028003",
        "Score": Float,
        "Description": "Ear, nose and throat examination
(procedure)"
    }
],
"Score": Float,
"RelationshipType": "SYSTEM_ORGAN_SITE",
"Type": "SYSTEM_ORGAN_SITE",
"Id": 0,
"BeginOffset": 0
}
],
"Type": "DX_NAME",
"Id": 4
}
],
"SNOMEDCTDetails": {
    "Edition": "US",
    "VersionDate": "20200901",
    "Language": "en"
},
"Characters": {
    "OriginalTextCharacters": 59
},
"ModelVersion": "3.2.0.20220301"

```

}

オントロジーリンクのバッチ分析

Amazon Comprehend Medical を使用して、Amazon Simple Storage Service (Amazon S3) バケットに保存されている臨床テキスト内のエンティティを検出し、これらのエンティティを標準化されたオントロジーにリンクします。オントロジーリンクのバッチ分析を使用して、ドキュメントのコレクションまたは最大 20,000 文字の単一ドキュメントを分析できます。コンソール、またはオントロジーリンクバッチ API オペレーションを使用して、継続中のバッチ分析ジョブの開始、停止、リスト表示、および説明を行うオペレーションを実行できます。

バッチ分析およびその他の Amazon Comprehend Medical オペレーションの価格情報については、「[Amazon Comprehend Medical の料金](#)」を参照してください。

バッチ分析の実行

バッチ分析ジョブは、Amazon Comprehend Medical コンソールまたは Amazon Comprehend Medical バッチ API オペレーションを使用して実行できます。

API オペレーションを使用したバッチ分析の実行

前提条件

Amazon Comprehend Medical API を使用する場合、AWS Identity Access and Management (IAM) ポリシーを作成し、このポリシーを IAM ロールにアタッチします。IAM ロールと信頼ポリシーの詳細については、「[IAM でのポリシーとアクセス許可](#)」を参照してください。

1. データを S3 バケットにアップロードします。
2. 新しい分析ジョブを開始するには、StartICD10CMInferenceJob オペレーション、StartSNOMEDCTInferenceJob オペレーション、または StartRxNormInferenceJob オペレーションを使用します。入力ファイルが含まれる Amazon S3 バケットの名前と、出力ファイルを送信する Amazon S3 バケットの名前を指定します。
3. DescribeICD10CMInferenceJob オペレーション、DescribeSNOMEDCTInferenceJob オペレーション、または DescribeRxNormInferenceJob オペレーションを使用して、ジョブの進行状況を監視します。また、ListICD10CMInferenceJobs、ListSNOMEDCTInferenceJobs、および ListRxNormInferenceJobs を使用して、オントロジーリンクのすべてのバッチ分析ジョブのステータスを確認できます。

4. 進行中のジョブを停止する必要がある場合は、StopICD10CMInferenceJob、StopSNOMEDCTInferenceJob、または StopRxnNormInferenceJob を使用して分析を停止します。
5. 分析ジョブの結果を表示するには、ジョブの開始時に設定した出力 S3 バケットを確認します。

コンソールを使用したバッチ分析の実行

1. データを S3 バケットにアップロードします。
2. 新しい分析ジョブを開始するには、実行する分析のタイプを選択します。入力ファイルが含まれる S3 バケットの名前と、出力ファイルを送信する S3 バケットの名前を指定します。
3. 進行中のジョブのステータスを監視します。コンソールから、分析の開始日と終了日など、すべてのバッチ分析オペレーションとそのステータスを表示できます。
4. 分析ジョブの結果を表示するには、ジョブの開始時に設定した出力 S3 バケットを確認します。

バッチオペレーションの IAM ポリシー

Amazon Comprehend Medical バッチ API オペレーションを呼び出す IAM ロールには、入力ファイルと出力ファイルが含まれる S3 バケットへのアクセス権を付与するポリシーが必要です。また、IAM ロールには、Amazon Comprehend Medical サービスがロールを引き受けることができるようにする信頼関係も割り当てる必要があります。IAM ロールと信頼ポリシーの詳細については、「[IAM ロール](#)」を参照してください。

ロールには、以下のポリシーが必要です。

JSON

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

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

ロールには、以下の信頼関係が必要です。混乱した代理によるセキュリティの問題を防ぐために `aws:SourceAccount` および `aws:SourceArn` 条件キーを使用することをお勧めします。混乱した代理問題と AWS アカウントを保護する方法の詳細については、IAM [ドキュメントの混乱した代理問題](#)を参照してください。

JSON

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

```
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "account_id"
      },
      "ArnLike": {
        "aws:SourceArn": "arn:aws:comprehendmedical:us-
east-1:account_id:*"
      }
    }
  ]
}
```

バッチ分析の出力ファイル

Amazon Comprehend Medical は、バッチ内の入力ファイルごとに 1 つの出力ファイルを作成します。ファイルの拡張子は .out です。Amazon Comprehend Medical は、まず、*AwsAccountId-JobType-JobId* を名前にして出力 S3 バケット内にディレクトリを作成し、バッチのすべての出力ファイルをこのディレクトリに書き込みます。Amazon Comprehend Medical は、ジョブの出力が別のジョブの出力を上書きしないように、この新しいディレクトリを作成します。

バッチオペレーションでは、同期オペレーションと同じ出力が生成されます。

バッチオペレーションごとに、ジョブに関する情報が含まれた以下の 3 つのマニフェストファイルが生成されます。

- Manifest — ジョブを要約します。ジョブに使用されたパラメータ、ジョブの合計サイズ、および処理されたファイル数に関する情報を提供します。
- Success — 処理が成功したファイルに関する情報を提供します。入力ファイル名と出力ファイル名、および入力ファイルのサイズが含まれます。
- Unprocessed — バッチジョブで処理されなかったファイルを、ファイルごとのエラーコードとエラーメッセージと共にリスト表示します。

Amazon Comprehend Medical は、バッチジョブ用に指定された出力ディレクトリにこれらのファイルを書き込みます。概要マニフェストファイルは、Manifest_AccountId-Operation-JobId というタイトルのフォルダと共に出力フォルダに書き込まれます。マニフェストフォルダには、成功マ

ニフェストを含む success フォルダと、未処理のファイルマニフェストを含む failed フォルダがあります。以降のセクションでは、マニフェストファイルの構造を示します。

バッチのマニフェストファイル

以下に、バッチのマニフェストファイルの JSON 構造を示します。

```
{
  "Summary" :
  {
    "Status" : "COMPLETED | FAILED | PARTIAL_SUCCESS | STOPPED",
    "JobType" : "ICD10CMInference | RxNormInference | SNOMEDCTInference",
    "InputDataConfiguration" : {
      "Bucket" : "input bucket",
      "Path" : "path to files/account ID-job type-job ID"
    },
    "OutputDataConfiguration" : {
      "Bucket" : "output bucket",
      "Path" : "path to files"
    },
    "InputFileCount" : number of files in input bucket,
    "TotalMeteredCharacters" : total characters processed from all files,
    "UnprocessedFilesCount" : number of files not processed,
    "SuccessFilesCount" : total number of files processed,
    "TotalDurationSeconds" : time required for processing,
    "SuccessfulFilesListLocation" : "path to file",
    "UnprocessedFilesListLocation" : "path to file",
    "FailedJobErrorMessage": "error message or if not applicable,
      The status of the job is completed"
  }
}
```

成功のマニフェストファイル

以下に、処理が成功したファイルに関する情報が含まれたファイルの JSON 構造を示します。

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

```
    ]]  
  }
```

未処理のマニフェストファイル

以下に、未処理のファイルに関する情報が含まれたマニフェストファイルの JSON 構造を示します。

```
{  
  "Files" : [ {  
    "Input": "file_name_that_failed",  
    "ErrorCode": "error code for exception",  
    "ErrorMessage": "explanation of the error code and suggestions"  
  },  
  { ... }  
 ]  
}
```

Amazon Comprehend Medical でのセキュリティ

のクラウドセキュリティが最優先事項 AWS です。お客様は AWS、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャを活用できます。

セキュリティは、AWS とお客様の間の責任共有です。[責任共有モデル](#)では、これをクラウドのセキュリティおよびクラウド内のセキュリティと説明しています。

- クラウドのセキュリティ – クラウドで AWS AWS サービスを実行するインフラストラクチャを保護する AWS 責任があります。AWS また、では、安全に使用できるサービスも提供しています。サードパーティーの監査者は、[AWS コンプライアンスプログラム](#)コンプライアンスプログラムの一環として、当社のセキュリティの有効性を定期的にテストおよび検証。Amazon Comprehend Medical に適用されるコンプライアンスプログラムの詳細については、「[コンプライアンスプログラムによる AWS 対象範囲内のサービス](#)」「」を参照してください。
- クラウドのセキュリティ – お客様の責任は、使用する AWS サービスによって決まります。また、ユーザーは、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、Comprehend Medical を使用する際に責任共有モデルを適用する方法を理解するのに役立ちます。以下のトピックでは、セキュリティおよびコンプライアンスの目的を達成するように Comprehend Medical を設定する方法について説明します。また、Comprehend Medical リソースのモニタリングや保護に役立つ他の AWS サービスを使用する方法についても説明します。

トピック

- [Amazon Comprehend Medical でのデータ保護](#)
- [Amazon Comprehend Medical でのアイデンティティおよびアクセスの管理](#)
- [を使用した Amazon Comprehend Medical API コールのログ記録 AWS CloudTrail](#)
- [Amazon Comprehend Medical のコンプライアンス検証](#)
- [Amazon Comprehend Medical での耐障害性](#)
- [Amazon Comprehend Medical でのインフラストラクチャセキュリティ](#)

Amazon Comprehend Medical でのデータ保護

責任 AWS [共有モデル](#)、Amazon Comprehend Medical でのデータ保護に適用されます。このモデルで説明されているように、AWS はすべての を実行するグローバルインフラストラクチャを保護す

る責任があります AWS クラウド。ユーザーは、このインフラストラクチャでホストされるコンテンツに対する管理を維持する責任があります。また、使用する「AWS のサービス」のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、[データプライバシーに関するよくある質問](#)を参照してください。欧州でのデータ保護の詳細については、AWS セキュリティブログに投稿された「[AWS 責任共有モデルおよび GDPR](#)」のブログ記事を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント、AWS IAM アイデンティティセンターまたは AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 は必須ですが、TLS 1.3 を推奨します。
- API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。CloudTrail 証跡を使用して AWS アクティビティをキャプチャする方法については、「AWS CloudTrail ユーザーガイド」の[CloudTrail 証跡の使用](#)を参照してください。
- AWS 暗号化ソリューションと、その中のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度な管理されたセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは API AWS を介して にアクセスするときに FIPS 140-3 検証済み暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「[連邦情報処理規格 \(FIPS\) 140-3](#)」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報を、タグ、または [名前] フィールドなどの自由形式のテキストフィールドに含めないことを強くお勧めします。これは、コンソール、API、または SDK を使用して Comprehend Medical AWS CLI または他の AWS のサービスを使用する場合も同様です。AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

Amazon Comprehend Medical でのアイデンティティおよびアクセスの管理

Amazon Comprehend Medical へのアクセスには、AWS によってリクエストの認証に使用される認証情報が必要です。これらの認証情報には、Comprehend Medical アクションにアクセスするためのアクセス許可が必要です。[AWS Identity and Access Management \(IAM\)](#) を使用すると、リソースにアクセスできるユーザーを制御してリソースを保護できます。以下のセクションでは、Comprehend Medical で IAM を使用する方法について詳しく説明します。

- [認証](#)
- [アクセスコントロール](#)

認証

Amazon Comprehend Medical を操作するためのアクセス許可をユーザーに付与する必要があります。フルアクセスが必要なユーザーの場合は、`ComprehendMedicalFullAccess` を使用します。

アクセス権限を付与するにはユーザー、グループ、またはロールにアクセス許可を追加します。

- 以下のユーザーとグループ AWS IAM アイデンティティセンター:

アクセス許可セットを作成します。「AWS IAM アイデンティティセンター ユーザーガイド」の「[アクセス許可セットを作成する](#)」の手順に従ってください。

- IAM 内で、ID プロバイダーによって管理されているユーザー:

ID フェデレーションのロールを作成します。詳細については IAM ユーザーガイドの [サードパーティー ID プロバイダー \(フェデレーション\) 用のロールを作成する](#) を参照してください。

- IAM ユーザー:

- ユーザーが担当できるロールを作成します。手順については IAM ユーザーガイドの [IAM ユーザーのロールの作成](#) を参照してください。

- (お奨めできない方法) ポリシーをユーザーに直接アタッチするか、ユーザーをユーザーグループに追加します。IAM ユーザーガイドの [ユーザー \(コンソール\) へのアクセス許可の追加](#) の指示に従います。

Amazon Comprehend Medical の非同期オペレーションを使用するには、サービスロールも必要です。

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#)です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、IAM ユーザーガイドの [AWS のサービスに許可を委任するロールを作成する](#)を参照してください。

Amazon Comprehend Medical をプリンシパルのサービスとして指定する方法については、「[バッチ操作に必要なロールベースのアクセス許可](#)」を参照してください。

アクセスコントロール

リクエストを認証できる有効な認証情報を持っている必要があります。認証情報には、Amazon Comprehend Medical アクションを呼び出すためのアクセス許可が必要です。

以下のセクションでは、Amazon Comprehend Medical のアクセス許可を管理する方法について説明します。最初に概要のセクションを読むことをお勧めします。

- [Amazon Comprehend Medical リソースに対するアクセス許可の管理の概要](#)
- [Amazon Comprehend Medical でのアイデンティティベースのポリシー \(IAM ポリシー\) の使用](#)

トピック

- [Amazon Comprehend Medical リソースに対するアクセス許可の管理の概要](#)
- [Amazon Comprehend Medical でのアイデンティティベースのポリシー \(IAM ポリシー\) の使用](#)
- [Amazon Comprehend Medical API のアクセス許可: アクション、リソース、条件のリファレンス](#)
- [AWS Amazon Comprehend Medical の マネージドポリシー](#)

Amazon Comprehend Medical リソースに対するアクセス許可の管理の概要

アクションへのアクセスは、アクセス許可ポリシーによって管理されます。アカウント管理者は、IAM アイデンティティにアクセス許可ポリシーをアタッチして、アクションへのアクセスを管理します。IAM アイデンティティには、ユーザー、グループ、およびロールが含まれます。

Note

アカウント管理者 (または管理者ユーザー) は、管理者権限を持つユーザーです。詳細については、「IAM ユーザーガイド」の「[IAM のベストプラクティス](#)」を参照してください。

アクセス許可を付与するときは、アクセス許可を付与するユーザーとアクションの両方を決定します。

トピック

- [アクションへのアクセスの管理](#)
- [ポリシー要素 \(アクション、効果、プリンシパル\) の指定](#)
- [ポリシーの条件の指定](#)

アクションへのアクセスの管理

アクセスポリシーは、誰が何に対するアクセス権を持っているのかを説明します。以下のセクションで、アクセス許可ポリシーのオプションについて説明します。

Note

このセクションでは、Amazon Comprehend Medical のコンテキストで IAM について説明します。ここでは、IAM サービスに関する詳細情報を提供しません。IAM の詳細については、IAM ユーザーガイドの「[IAM とは](#)」を参照してください。IAM ポリシー構文の詳細と説明については、「IAM ユーザーガイド」の「[AWS IAM ポリシーリファレンス](#)」を参照してください。

IAM アイデンティティにアタッチされたポリシーは、アイデンティティベースのポリシーです。リソースにアタッチされたポリシーは、リソースベースのポリシーです。Amazon Comprehend Medical では、アイデンティティベースのポリシーのみがサポートされています。

アイデンティティベースのポリシー (IAM ポリシー)

ポリシーを IAM アイデンティティにアタッチできます。これらはその 2 つの例です。

- アカウントのユーザーまたはグループにアクセス許可ポリシーをアタッチする。ユーザーまたはユーザーのグループが Amazon Comprehend Medical アクションを呼び出すことができるようにするには、アクセス許可ポリシーをユーザーにアタッチします。ユーザーが含まれているグループにポリシーをアタッチします。
- ロールにアクセス許可ポリシーをアタッチし、クロスアカウントのアクセス許可を付与する。クロスアカウントのアクセス許可を付与するには、アイデンティティベースのポリシーを IAM ロー

ルにアタッチします。例えば、アカウント A の管理者は、別のアカウントにクロスアカウントのアクセス許可を付与するロールを作成できます。この例では、これをアカウント B と呼んでおり、AWS サービスにすることもできます。

1. アカウント A の管理者は、IAM ロールを作成し、そのロールに、アカウント A のリソースに対するアクセス許可を付与するポリシーをアタッチします。
2. アカウント A の管理者は、ロールに信頼ポリシーをアタッチします。ポリシーは、ロールを引き受けることのできるプリンシパルとしてアカウント B を識別します。
3. アカウント B の管理者は、アカウント B の任意のユーザーにロールを引き受けるアクセス許可を委任できるようになります。これにより、アカウント B のユーザーにアカウント A のリソースの作成またはアクセスが許可されます。ロールを引き受けるアクセス許可を AWS のサービスに付与すると、信頼ポリシーのプリンシパルは AWS のサービスのプリンシパルになることもできます。

IAM を使用した許可の委任の詳細については、「IAM ユーザーガイド」の「[アクセス管理](#)」を参照してください。

Amazon Comprehend Medical でアイデンティティベースのポリシーを使用する場合の詳細については、「[Amazon Comprehend Medical でのアイデンティティベースのポリシー \(IAM ポリシー\) の使用](#)」を参照してください。ユーザー、グループ、ロール、許可の詳細については、「[IAM ユーザーガイド](#)」の「アイデンティティ (ユーザー、グループ、ロール)」を参照してください。

リソースベースのポリシー

などの他の サービスは AWS Lambda、リソースベースのアクセス許可ポリシーをサポートしています。例えば、ポリシーを S3 バケットにアタッチして、そのバケットに対するアクセス許可を管理できます。Amazon Comprehend Medical では、リソースベースのポリシーはサポートされていません。

ポリシー要素 (アクション、効果、プリンシパル) の指定

Amazon Comprehend Medical では、一連の API オペレーションを定義します。これらの API オペレーションへのアクセス許可を付与するために、Amazon Comprehend Medical では、ポリシーに指定できる一連のアクションを定義します。

以下の 4 つの項目は、最も基本的なポリシー要素です。

- リソース – ポリシーで Amazon リソースネーム (ARN) を使用して、ポリシーを適用するリソースを識別します。Amazon Comprehend Medical の場合、リソースは常に "*" です。

- アクション – アクションのキーワードを使用して、許可または拒否するオペレーションを識別します。例えば、指定した効果に応じて、comprehendmedical:DetectEntities は Amazon Comprehend Medical DetectEntities オペレーションを実行するためのアクセス許可をユーザーに付与または拒否します。
- 効果 - ユーザーが特定のアクションをリクエストしたときに起きるアクションの効果 (許可または拒否のいずれか) を指定します。リソースへのアクセスを明示的に許可していない場合、アクセスは暗黙的に拒否されます。リソースへのアクセスを明示的に拒否することもできます。これにより、別のポリシーでアクセスが許可されている場合でも、ユーザーがリソースにアクセスすることを禁止できます。
- プリンシパル - アイデンティティベースのポリシーで、ポリシーがアタッチされているユーザーが默示のプリンシパルになります。

IAM ポリシーの構文と記述の詳細については、「IAM ユーザーガイド」の「[AWS IAM ポリシーリファレンス](#)」を参照してください。

すべての Amazon Comprehend Medical API アクションを示す表については、「[Amazon Comprehend Medical API のアクセス許可: アクション、リソース、条件のリファレンス](#)」を参照してください。

ポリシーの条件の指定

アクセス許可を付与するときに、IAM ポリシー言語を使用してポリシーの適用条件を指定します。例えば、特定の日付の後にのみ適用されるポリシーが必要になる場合があります。ポリシー言語での条件の指定の詳細については、「IAM ユーザーガイド」の「[条件](#)」を参照してください。

AWS では、IAM をサポートする AWS のすべてのサービスについて、アクセスコントロール用の条件キーのセットが事前に定義されています。たとえば、aws:userid 条件キーを使用して、アクションのリクエスト時に特定の AWS ID を要求できます。AWS キーの詳細およびすべての AWS キーのリストについては、IAM ユーザーガイドの「[使用可能な条件キー](#)」を参照してください。

Amazon Comprehend Medical では、追加の条件キーが提供されません。

Amazon Comprehend Medical でのアイデンティティベースのポリシー (IAM ポリシー) の使用

このトピックでは、アイデンティティベースのポリシーの例を示します。例を使用して、アカウント管理者が IAM アイデンティティにアクセス許可ポリシーをアタッチする方法を示します。これによ

り、ユーザー、グループ、およびロールが Amazon Comprehend Medical アクションを実行できるようになります。

⚠ Important

アクセス許可を理解するには、「[Amazon Comprehend Medical リソースに対するアクセス許可の管理の概要](#)」をお勧めします。

このポリシー例は、Amazon Comprehend Medical ドキュメント分析アクションを使用するために必要です。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowDetectActions",
      "Effect": "Allow",
      "Action": [
        "comprehendmedical:DetectEntitiesV2",
        "comprehendmedical:DetectPHI",
        "comprehendmedical:StartEntitiesDetectionV2Job",
        "comprehendmedical:ListEntitiesDetectionV2Jobs",
        "comprehendmedical:DescribeEntitiesDetectionV2Job",
        "comprehendmedical:StopEntitiesDetectionV2Job",
        "comprehendmedical:StartPHIDetectionJob",
        "comprehendmedical:ListPHIDetectionJobs",
        "comprehendmedical:DescribePHIDetectionJob",
        "comprehendmedical:StopPHIDetectionJob",
        "comprehendmedical:StartRxNormInferenceJob",
        "comprehendmedical:ListRxNormInferenceJobs",
        "comprehendmedical:DescribeRxNormInferenceJob",
        "comprehendmedical:StopRxNormInferenceJob",
        "comprehendmedical:StartICD10CMInferenceJob",
        "comprehendmedical:ListICD10CMInferenceJobs",
        "comprehendmedical:DescribeICD10CMInferenceJob",
        "comprehendmedical:StopICD10CMInferenceJob",
        "comprehendmedical:StartSNOMEDCTInferenceJob",
        "comprehendmedical:ListSNOMEDCTInferenceJobs",

```

```

        "comprehendmedical:DescribeSNOMEDCTInferenceJob",
        "comprehendmedical:StopSNOMEDCTInferenceJob",
        "comprehendmedical:InferRxNorm",
        "comprehendmedical:InferICD10CM",
        "comprehendmedical:InferSNOMEDCT"
    ],
    "Resource": "*"
}
]
}

```

このポリシーには、DetectEntities アクションおよび DetectPHI アクションを使用するためのアクセス許可を付与するステートメントが 1 つあります。

アイデンティティベースのポリシーでアクセス権限を得るプリンシパルを指定していないため、ポリシーでは Principal エlement を指定していません。ユーザーにポリシーをアタッチすると、そのユーザーが暗黙のプリンシパルになります。IAM ロールにポリシーをアタッチすると、ロールの信頼ポリシーで識別されたプリンシパルがアクセス許可を得ることになります。

すべての Amazon Comprehend Medical API アクションとそれらが適用されるリソースを確認するには、「[Amazon Comprehend Medical API のアクセス許可: アクション、リソース、条件のリファレンス](#)」を参照してください。

Amazon Comprehend Medical コンソールを使用するために必要なアクセス許可

アクセス許可のリファレンス表では、Amazon Comprehend Medical API オペレーションとそれらの各オペレーションに必要なアクセス許可を示しています。Amazon Comprehend Medical API のアクセス許可の詳細については、「[Amazon Comprehend Medical API のアクセス許可: アクション、リソース、条件のリファレンス](#)」を参照してください。

Amazon Comprehend Medical コンソールを使用するには、次のポリシーに示されているアクションのアクセス許可を付与する必要があります。

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {

```

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

Amazon Comprehend Medical コンソールには、以下の理由でこれらのアクセス許可が必要になります。

- アカウントで使用可能な IAM ロールをリストするための iam アクセス許可。
- データが含まれる Amazon S3 バケットおよびオブジェクトにアクセスするための s3 アクセス許可。

コンソールを使用して非同期バッチジョブを作成する場合、ジョブに IAM ロールを作成することもできます。コンソールを使用して IAM ロールを作成するには、IAM ロールとポリシーを作成してロールにポリシーをアタッチするために、ここに示されている追加のアクセス許可をユーザーに付与する必要があります。

JSON

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

```
"Statement": [  
  {  
    "Action": [  
      "iam:CreateRole",  
      "iam:CreatePolicy",  
      "iam:AttachRolePolicy"  
    ],  
    "Effect": "Allow",  
    "Resource": "*"   
  }  
]
```

Amazon Comprehend Medical コンソールでは、ロールとポリシーを作成してロールとポリシーをアタッチするには、これらのアクセス許可が必要です。iam:PassRole アクションによって、コンソールで Amazon Comprehend Medical にロールを渡すことができます。

Amazon Comprehend Medical の AWS 管理 (事前定義) ポリシー

AWS は、AWS によって作成され管理されるスタンドアロンの IAM ポリシーが提供する多くの一般的なユースケースに対応します。これらの AWS 管理ポリシーは、一般的ユースケースに必要なアクセス許可を付与することで、どのアクセス許可が必要なのかをユーザーが調査する必要をなくすことができます。詳細については、[「IAM ユーザーガイド」](#)の「AWS マネージドポリシー」を参照してください。

アカウントのユーザーにアタッチ可能な以下の AWS 管理ポリシーは、Amazon Comprehend Medical に固有のものです。

- ComprehendMedicalFullAccess — Amazon Comprehend Medical リソースへのフルアクセスを付与します。IAM ロールをリストおよび取得するアクセス許可が含まれます。

Amazon Comprehend Medical を使用するユーザーには、次の追加ポリシーを適用する必要があります。

JSON

```
{
```

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

IAM コンソールにサインインし、特定のポリシーを検索することで、管理アクセス許可ポリシーを確認できます。

これらのポリシーは、AWS SDK または AWS CLI を使用しているときに機能します。

独自の IAM ポリシーを作成して、Amazon Comprehend Medical のアクションとリソースのためのアクセス許可を付与することもできます。これらのカスタムポリシーを、これらのポリシーを必要とする IAM ユーザーまたはグループにアタッチできます。

バッチ操作に必要なロールベースのアクセス許可

Amazon Comprehend Medical 非同期オペレーションを使用するには、Amazon Comprehend Medical にドキュメントコレクションが含まれる Amazon S3 バケットへのアクセス許可を付与します。これを行うには、Amazon Comprehend Medical サービスプリンシパルを信頼するように、アカウントにデータアクセスロールを作成します。ロールの作成について詳しくは、AWS Identity and Access Management ユーザーガイドの「[AWS のサービスにアクセス許可を委任するロールの作成](#)」を参照してください。

以下に示しているのは、ロールの信頼ポリシーです。

JSON

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

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

ロールを作成したら、そのロールのアクセスポリシーを作成します。ポリシーによって、入力データが含まれる Amazon S3 バケットへの Amazon S3 GetObject アクセス許可および ListBucket アクセス許可を付与する必要があります。また、Amazon S3 出力データバケットへの Amazon S3 PutObject のアクセス許可も付与します。

次のアクセスポリシー例には、これらのアクセス許可が含まれています。

JSON

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

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

カスタマーマネージドポリシーの例

このセクションでは、さまざまな Amazon Comprehend Medical アクションのアクセス許可を付与するユーザーポリシー例を示しています。これらのポリシーは、AWS SDK または AWS CLI を使用しているときに機能します。コンソールを使用している場合、すべての Amazon Comprehend Medical API にアクセス許可を付与する必要があります。これについては、「[Amazon Comprehend Medical コンソールを使用するために必要なアクセス許可](#)」を参照してください。

Note

すべての例で、us-east-2 リージョンを使用し、架空のアカウント ID を含めています。

例

例 1: すべての Amazon Comprehend Medical アクションを許可する

にサインアップしたら AWS、ユーザーの作成やアクセス許可の管理など、アカウントを管理する管理者を作成します。

すべての Amazon Comprehend アクションのアクセス許可を持つユーザーを作成できます。このユーザーは、Amazon Comprehend を使用するためのサービス固有の管理者とを考えてください。このユーザーに以下のアクセス権限をアタッチできます。

JSON

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

例 2: DetectEntities アクションのみを許可する

次のアクセス許可ポリシーでは、Amazon Comprehend Medical でエンティティを検出するためのアクセス許可がユーザーに付与されますが、PHI オペレーションを検出するためのアクセス許可は付与されません。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowDetectEntityActions",
      "Effect": "Allow",
      "Action": [
        "comprehendmedical:DetectEntitiesV2"
      ],
      "Resource": "*"
    }
  ]
}
```

Amazon Comprehend Medical API のアクセス許可: アクション、リソース、条件のリファレンス

以下の表は、[アクセスコントロール](#)をセットアップし、ユーザーにアタッチできる許可ポリシーを作成するときのリファレンスとして使用してください。この表は、各 Amazon Comprehend Medical API オペレーション、アクションを実行するためのアクセス許可を付与できる対象のアクション、およびアクセス許可を付与できる AWS リソースを示しています。ポリシーの Action フィールドでアクションを指定し、ポリシーの Resource フィールドでリソースの値を指定します。

条件を表すために、Amazon Comprehend Medical ポリシーで AWS の条件キーを使用できます。すべてのキーのリストについては、IAM ユーザーガイドの「[利用可能なキー](#)」を参照してください。

Note

アクションを指定するには、comprehendmedical: プレフィックスに続けて API オペレーション名を使用します (例: comprehendmedical:DetectEntities)。

AWS Amazon Comprehend Medical の マネージドポリシー

AWS 管理ポリシーは、によって作成および管理されるスタンドアロンポリシーです AWS。AWS 管理ポリシーは、多くの一般的なユースケースにアクセス許可を付与するように設計されているため、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できます。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケースに固有の[カスタマー管理ポリシー](#)を定義して、アクセス許可を絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS 管理ポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) に影響します。AWS は、新しい が起動されるか、新しい API オペレーション AWS のサービス が既存のサービスで使用できるようになったときに、AWS 管理ポリシーを更新する可能性が高くなります。

詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」を参照してください。

トピック

- [AWS マネージドポリシー: ComprehendMedicalFullAccess](#)
- [AWS 管理ポリシーに対する Comprehend Medical の更新](#)

AWS マネージドポリシー: ComprehendMedicalFullAccess

ComprehendMedicalFullAccess ポリシーを IAM アイデンティティにアタッチできます。

このポリシーは、すべての Comprehend Medical アクションに管理アクセス許可を付与します。

JSON

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

AWS 管理ポリシーに対する Comprehend Medical の更新

このサービスがこれらの変更の追跡を開始してからの Amazon Comprehend Medical の AWS マネージドポリシーの更新に関する詳細を表示します。このページの変更に関する自動通知については、「[ドキュメント履歴ページ](#)」ページで RSS フィードをサブスクライブしてください。

変更	説明	日付
Comprehend Medical が変更の追跡を開始	Comprehend Medical は、AWS 管理ポリシーの変更の追跡を開始しました。	2018 年 11 月 27 日

を使用した Amazon Comprehend Medical API コールのログ記録 AWS CloudTrail

Amazon Comprehend Medical はと統合されています AWS CloudTrail。CloudTrail は、Amazon Comprehend Medical 内からユーザー、ロール、または AWS のサービスによって実行されたアクションを記録するサービスです。CloudTrail は、Amazon Comprehend Medical のすべての API コールをイベントとしてキャプチャします。キャプチャされるコールには、Amazon Comprehend Medical コンソールからのコールと、Amazon Comprehend Medical API オペレーションへのコードコールが含まれます。証跡を作成する場合は、Amazon Comprehend Medical のイベントを含めた CloudTrail イベントの Amazon S3 バケットへの継続的な配信を有効にすることができます。証跡を設定しない場合でも、CloudTrail コンソールの [イベント履歴] で最新のイベントを表示できます。CloudTrail によって収集された情報を使用して、以下のような多くのことを判断できます。

- Amazon Comprehend Medical に対して行われたリクエスト
- リクエストが行われた IP アドレス
- リクエストを行ったユーザー
- リクエストが行われた時間
- その他の詳細

CloudTrail の詳細については、「[AWS CloudTrail ユーザーガイド](#)」を参照してください。

CloudTrail での Amazon Comprehend Medical 情報

CloudTrail は、AWS アカウントの作成時にアカウントで有効になります。Amazon Comprehend Medical でアクティビティが発生すると、そのアクティビティはイベント履歴の他の AWS サービス イベントとともに CloudTrail イベントに記録されます。AWS アカウントで最近のイベントを表示、検索、ダウンロードできます。詳細については、「[CloudTrail イベント履歴でのイベントの表示](#)」を参照してください。

Amazon Comprehend Medical のイベントなど、AWS アカウント内のイベントの継続的な記録については、証跡を作成します。追跡により、CloudTrailはログファイルをSimple Storage Service (Amazon S3) バケットに配信できます。デフォルトでは、コンソールで作成した証跡がすべてのAWS リージョンに適用されます。証跡は、AWS パーティション内のすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集されたイベントデータをさらに分析して処理するように、他の AWS サービスを設定できます。詳細については、次を参照してください:

- [証跡の作成のための概要](#)
- [CloudTrail がサポートするサービスと統合](#)
- [CloudTrail 用 Amazon SNS 通知の構成](#)
- 「[複数のリージョンからCloudTrailログファイルを受け取る](#)」および「[複数のアカウントからCloudTrailログファイルを受け取る](#)」

すべての Amazon Comprehend Medical アクションは、CloudTrail によってログに記録され、[Amazon Comprehend Medical API リファレンス](#)に記載されています。例えば、DetectEntitiesV2、DetectPHI、ListEntitiesDetectionV2Jobs の各アクションを呼び出すと、CloudTrail ログファイルにエントリが生成されます。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストが root または AWS Identity and Access Management (IAM) ユーザー認証情報を使用して行われたかどうか。
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが別の AWS サービスによって行われたかどうか。

詳細については、[CloudTrail userIdentity 要素](#)を参照してください。

Amazon Comprehend Medical ログファイルエントリについて

「トレイル」は、指定した Amazon S3 バケットにイベントをログファイルとして配信するように設定できます。CloudTrail ログファイルには、1 つ以上のログエントリがあります。イベントは、任意の送信元からの単一の要求を表します。イベントには、日時やリクエストのパラメータなど、要求されたアクションに関する情報が含まれます。CloudTrail ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

以下の例は、DetectEntitiesV2 アクションを示す CloudTrail ログエントリです。

```
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/Mateo_Jackson",
        "accountId": "123456789012",
        "accessKeyId": "ASIAXHKUFODNN8EXAMPLE",
        "sessionContext": {
          "sessionIssuer": {
            "type": "Role",
            "principalId": "AIDACKCEVSQ6C2EXAMPLE",
            "arn": "arn:aws:iam::123456789012:user/Mateo_Jackson",
            "accountId": "123456789012",
            "userName": "Mateo_Jackson"
          },
          "webIdFederationData": {},
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2019-09-27T20:07:27Z"
          }
        }
      },
      "eventTime": "2019-09-27T20:10:26Z",
      "eventSource": "comprehendmedical.amazonaws.com",
      "eventName": "DetectEntitiesV2",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "702.21.198.166",
      "userAgent": "aws-internal/3 aws-sdk-java/1.11.590
Linux/4.9.184-0.1.ac.235.83.329.metal1.x86_64 OpenJDK_64-Bit_Server_VM/25.212-b03
java/1.8.0_212 vendor/Oracle_Corporation",
      "requestParameters": null,
      "responseElements": null,
      "requestID": "8d85f2ec-EXAMPLE",
      "eventID": "ae9be9b1-EXAMPLE",
      "eventType": "AwsApiCall",
      "recipientAccountId": "123456789012"
    }
  }
```

Amazon Comprehend Medical のコンプライアンス検証

サードパーティーの監査者は、複数のコンプライアンスプログラムの一環として Amazon Comprehend Medical のセキュリティと AWS コンプライアンスを評価します。このプログラムには、PCI、FedRAMP、HIPAA などがあります。を使用して、サードパーティーの監査レポートをダウンロードできます AWS Artifact。詳細については、[「Downloading Reports in AWS Artifact」](#) (AWS Artifact のレポートのダウンロード) を参照してください。

Comprehend Medical を使用する場合のお客様のコンプライアンス責任は、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。は、コンプライアンスに役立つ以下のリソース AWS を提供します。

- [セキュリティとコンプライアンスのクイックスタートガイド](#) – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、機密性とコンプライアンスに焦点を当てたベースライン環境を AWS にデプロイするためのステップを提供します。
- [「Architecting for HIPAA Security and Compliance」ホワイトペーパー](#) – このホワイトペーパーでは、企業が AWS を使用して HIPAA 準拠のアプリケーションを作成する方法について説明します。
- [AWS コンプライアンスリソース](#) – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- [AWS Config](#) – この AWS サービスは、リソース設定が内部プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。
- [AWS Security Hub CSPM](#) – この AWS サービスは、内のセキュリティ状態を包括的に把握 AWS し、セキュリティ業界標準とベストプラクティスへの準拠を確認するのに役立ちます。

特定のコンプライアンスプログラムの対象となる AWS サービスのリストについては、[「コンプライアンスプログラムによる AWS 対象範囲内のサービス」](#) を参照してください。一般的な情報については、[「AWS コンプライアンスプログラム」](#) を参照してください。

Amazon Comprehend Medical での耐障害性

AWS グローバルインフラストラクチャは、AWS リージョンとアベイラビリティゾーンを中心に構築されています。AWS リージョンは、低レイテンシー、高スループット、高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティゾーンを提供します。アベイラビリティゾーンでは、アベイラビリティゾーン間で中断せずに、自動的にフェイル

オーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性、耐障害性、およびスケーラビリティが優れています。

AWS リージョンとアベイラビリティゾーンの詳細については、[AWS 「グローバルインフラストラクチャ」](#) を参照してください。

Amazon Comprehend Medical でのインフラストラクチャセキュリティ

マネージドサービスである Amazon Comprehend Medical は、ホワイトペーパー「[Amazon Web Services: セキュリティプロセスの概要](#)」に記載されている AWS グローバルネットワークセキュリティ手順で保護されています。

ネットワーク経由で Comprehend Medical にアクセスするには、AWS 公開された API コールを使用します。クライアントで Transport Layer Security (TLS) 1.0 以降がサポートされている必要があります。TLS1.2以降が推奨されています。また、Ephemeral Diffie-Hellman (DHE) や Elliptic Curve Ephemeral Diffie-Hellman (ECDHE) などの Perfect Forward Secrecy (PFS) を使用した暗号スイートもクライアントでサポートされている必要があります。これらのモードは、Java 7 以降など、最近のほとんどのシステムでサポートされています。

さらに、(AWS Identity and Access Management IAM) プリンシパルに関連付けられたアクセスキー ID とシークレットアクセスキーを使用してリクエストに署名する必要があります。または、[AWS Security Token Service](#) (AWS STS) を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

ガイドラインとクォータ

Amazon Comprehend Medical を使用するときには、次の点に注意してください。

Note

- Amazon Comprehend Medical がサポートする文字エンコーディングは、UTF-8 英語 (EN) です。
- Amazon Comprehend Medical では、非同期ジョブのファイルパスでスラッシュ文字 (/) を順番に転送することはできません。

重要な注意点

Amazon Comprehend Medical は専門家による医療の助言、診断、治療の代用品ではありません。Amazon Comprehend Medical は、検出されたエンティティの精度に対する信頼度を示す信頼スコアを提供します。ユースケースに適した信頼しきい値を特定し、高い精度を必要とする状況では高い信頼しきい値を使用してください。特定のユースケースでは、適切な訓練を受けたレビュー担当者によって人的に結果を見直し、検証する必要があります。Amazon Comprehend Medical は、訓練を受けた医療専門家による結果の正確さと健全な医療判断の確認後に、患者ケアのシナリオでのみ使用してください。

サポートされるリージョン

Amazon Comprehend Medical が利用可能な AWS リージョンの一覧については、アマゾン ウェブ サービス全般リファレンスの [AWS リージョンとエンドポイント](#) を参照してください。

スロットリング

Amazon Comprehend Medical のスロットリングとクォータの詳細およびクォータの引き上げをリクエストする方法については、「[AWS Service Quotas](#)」を参照してください。

全体的なクォータ

Amazon Comprehend Medical リアルタイム (同期) 分析オペレーションには、以下のクォータがあります。

リソース	デフォルト
Infe EDCT オペレーションの1秒あたりのトランザクション	2 TPS

リ ソー ス	デ フォ ルト
--------------	---------------

DetectEnt 40,000
 DetectPHI CPS
 DetectNer v2, DetectPHI
 DetectSentiment DetectPHI
 DetectSyntax DetectPHI
 DetectICD10CM DetectPHI
 の
 各
 オ
 ペ
 レー
 ショ
 ンの
 1
 秒
 あ
 た
 り
 の
 文
 字
 数
 (CPS)

リソース	デフォルト
Infe EDCT オペレーション の 1 秒 あたり の 文字 数 (CPS	5,000 CPS

リ ソー ス	デ フォ ルト
--------------	---------------

Default 20
KB detectEnt

Default

v2、

お

よ

び

Default

オ

ペ

レー

ション

ン

の

最

大

ド

キュ

メ

ン

ト

サ

イ

ズ

(UTF-

文

字)。

リソース	デフォルト
Inference CM および Inference オペレーションの最大ドキュメントサイズ (UTF-8 文字)。	10 KB

リソース	デフォルト
Infe 5 EDCT KB オ ペ レー シヨ ン の 最 大 ド キュ メ ン ト サイ ズ (UTF 文 字)	

Amazon Comprehend Medical バッチ分析 (非同期) オペレーションには、以下のクォータがあります。

説明	クォータ
StartEntitiesDetectionV2Job 、 StartPHIDetectionJob 、 StopEntitiesDetectionV2Job 、 StopPHIDetectionJob 、 StartICD10CMInferenceJob 、 StartRxNormInferen	5 TPS

説明	クォータ
ceJob、StopICD10CMInferenceJob、StopRxNormInferenceJob、StartSNOMEDCTInferenceJob、StopSNOMEDCTInferenceJob の各オペレーションの 1 秒あたりのトランザクション数 (TPS)	
ListEntitiesDetectionV2Jobs、ListPHIDetectionJobs、DescribeEntitiesDetectionV2Job、DescribePHIDetectionJob、ListICD10CMInferenceJobs、ListRxNormInferenceJobs、DescribeICD10CMInferenceJob、DescribeRxNormInferenceJob、ListSNOMEDCTInferenceJobs、DescribeSNOMEDCTInferenceJob の各オペレーションの 1 秒あたりのトランザクション数 (TPS)	10 TPS
すべてのオペレーションのバッチジョブの最大個別ファイルサイズ	70 KB
バッチジョブ内の個々のファイルの最大数	5000000
バッチジョブの最大サイズ (バッチジョブで送信されたすべてのファイルの合計)。	1 GB
1 回のオペレーションで実行中のアクティブなバッチジョブの最大数	10 個のジョブ

テキストが文字クォータより大きい場合は、[segment.py](#) を使用して、分析できる小さいセグメントを作成します。

Amazon Comprehend Medical のドキュメント履歴

以下の表は、Amazon Comprehend Medical の今回のリリースのドキュメント内容をまとめたものです。

変更	説明	日付
InferSNOMEDCT の API バージョンの更新	InferSNOMEDCT API オペレーションでバージョンを使用するようになりました3.2.0.20220301。	2025 年 11 月 3 日
InferRxNorm の API バージョンの更新	InferRxNorm API オペレーションでバージョンを使用するようになりました3.2.0.20221107。	2025 年 11 月 3 日
InferICD10CM の API バージョンの更新	InferICD10CM API オペレーションでバージョンを使用するようになりました3.2.0.20251001。	2025 年 11 月 3 日
InferSNOMEDCT の API バージョンの更新	InferSNOMEDCT API オペレーションでバージョンを使用するようになりました3.1.0.20220301。	2025 年 8 月 5 日
InferRxNorm の API バージョンの更新	InferRxNorm API オペレーションでバージョンを使用するようになりました3.1.0.20221107。	2025 年 8 月 5 日
InferICD10CM の API バージョンの更新	InferICD10CM API オペレーションでバージョンを使用するようになりました3.1.0.20250401。	2025 年 8 月 5 日

InferSNOMEDCT の API バージョンの更新	InferSNOMEDCT API オペレーションでバージョンを使用するようになりました3.0.0.20220301。	2024年6月26日
InferRxNorm の API バージョンの更新	InferRxNorm API オペレーションでバージョンを使用するようになりました3.0.0.20221107。	2024年6月26日
InferICD10CM の API バージョンの更新	InferICD10CM API オペレーションでバージョンを使用するようになりました3.0.0.20231001。	2024年6月26日
DetectEntitiesV2 の API バージョンの更新	DetectEntitiesV2 API オペレーションでバージョンを使用するようになりました3.0.0。	2024年6月26日
DetectEntities の API バージョンの更新	DetectEntities API オペレーションでバージョンを使用するようになりました3.0.0。	2024年6月26日
InferICD10CM の API バージョンの更新	API オペレーション InferICD10CM はバージョン 2.7.0.20231001 を使用するようになりました。	2024年2月7日
InferICD10CM の API バージョンの更新	API オペレーション InferICD10CM はバージョン 2.6.0.20220401 を使用するようになりました。	2023年9月27日

InferSNOMEDCT の API バージョンの更新	API オペレーション InferSNOMEDCT は 3.2.0.20220301 を使用するようになりました。	2023 年 6 月 12 日
DetectEntitiesV2 に新しい特性と属性を追加	API オペレーション DetectEntitiesV2 はバージョン 3.0.0 を使用するようになりました。この更新により、新しい特性と属性が追加されました。	2023 年 6 月 12 日
DetectEntities に新しい特性と属性を追加	API オペレーション DetectEntities はバージョン 3.0.0 を使用するようになりました。この更新により、新しい特性と属性が追加されました。	2023 年 6 月 12 日
InferRxNorm に新しい特性を追加	API オペレーション InferRxNorm はバージョン 3.2.0.20221107 を使用するようになりました。この更新により、検出された薬剤に PAST_HISTORY 特性が追加されました。	2023 年 6 月 12 日
InferICD10CM に新しい属性を追加	API オペレーション InferICD10CM はバージョン 2.5.0.20220401 を使用するようになりました。この更新により、病状カテゴリに QUALITY 属性が追加されました。	2023 年 6 月 12 日

[InferRxNorm の API バージョ
ンの更新](#)

API オペレーション
InferRxNorm はバージョン
2.1.0.20221003 を使用す
るようになりました。この更
新により、InferRxNorm モデ
ルのパフォーマンスが向上し
ました。

2023 年 4 月 28 日

[InferSNOMEDCT の API バー
ジョンの更新](#)

API オペレーション
InferSNOMEDCT は
2.5.0.20220301 を使用す
るようになりました。この更
新により、モデルのパフォー
マンスが向上しました。

2023 年 4 月 21 日

[InferICD10CM の API バー
ジョンの更新](#)

API オペレーション
InferICD10CM はバージョ
ン 2.4.0.20220401 を使
用するようになりました。こ
の更新により、DX_NAME エン
ティティと DIRECTION 属性
の検出が向上しました。

2023 年 3 月 17 日

[InferSNOMEDCT の API バー
ジョンの更新](#)

API オペレーション
InferSNOMEDCT はバージョ
ン 2.4.0.20220301 を使
用するようになりました。こ
の更新により、MEDICAL_C
ONDITION および
TEST_TREATMENT_PRO
CEDURE カテゴリにおける心
臓病学の概念の検出が向上し
ました。

2023 年 3 月 15 日

[InferICD10CM の API バージョンの更新](#)

API オペレーション InferICD10CM はバージョン 2.3.0.20220401 を使用するようになりました。この更新により、MEDICAL_CONDITION カテゴリにおける心臓病学の概念の検出が向上しました。

2023 年 3 月 15 日

[DetectEntitiesV2 の API バージョンの更新](#)

API オペレーション DetectEntitiesV2 はバージョン 2.3.0 を使用するようになりました。この更新により、MEDICAL_CONDITION および TEST_TREATMENT_PROCEDURE カテゴリにおける心臓病学の概念の検出が向上しました。

2023 年 3 月 15 日

[DetectEntities の API バージョンの更新](#)

API オペレーション DetectEntities はバージョン 0.6.0 を使用するようになりました。この更新により、MEDICAL_CONDITION および TEST_TREATMENT_PROCEDURE カテゴリにおける心臓病学の概念の検出が向上しました。

2023 年 3 月 15 日

[InferSNOMEDCT の API バージョンの更新](#)

API オペレーション InferSNOMEDCT はバージョン 2.3.0.20220301 を使用するようになりました。この更新により、InferSNOMEDCT API オペレーションを使用する場合の否定の検出が向上します。

2023 年 2 月 8 日

[InferICD10CM の API バージョンの更新](#)

API オペレーション InferICD10CM はバージョン 2.2.0.20220401 を使用するようになりました。この更新により、InferICD10CM API オペレーションを使用する場合の否定の検出が向上します。

2023 年 2 月 8 日

[DetectEntitiesV2 の API バージョンの更新](#)

API オペレーション DetectEntitiesV2 はバージョン 2.2.0 を使用するようになりました。この更新により、DetectEntitiesV2 API オペレーションを使用する場合の否定の検出が向上します。

2023 年 2 月 8 日

[DetectEntities の API バージョンの更新](#)

API オペレーション DetectEntities はバージョン 0.5.0 を使用するようになりました。この更新により、DetectEntities API オペレーションを使用する場合の否定の検出が向上します。

2023 年 2 月 8 日

[InferSNOMEDCT の API バージョンの更新](#)

API オペレーション InferSNOMEDCT はバージョン 2.2.0.20220301 を使用するようになりました。この更新により、特定された属性の全セットが API 出力で返されないバグが修正されました。

2022 年 12 月 9 日

[InferSNOMEDCT の API バージョンの更新](#)

API オペレーション
InferSNOMEDCT はバージョン 2.1.0.20220301 を使用するようになりました。この更新により、精度が向上し、新しい特性が検出されるようになりました。

2022 年 11 月 16 日

[InferRxNorm の API バージョンの更新](#)

API オペレーション
InferRxNorm はバージョン 2.1.0.20221003 を使用するようになりました。この更新により、特定された属性の全セットが返されないバグが修正されました。

2022 年 11 月 16 日

[InferICD10CM の API バージョンの更新](#)

API オペレーション
InferICD10CM はバージョン 2.1.0.20220401 を使用するようになりました。この更新により、精度が向上し、新しい特性が検出されるようになりました。

2022 年 11 月 16 日

[DetectEntitiesV2 の API バージョンの更新](#)

API オペレーション
DetectEntitiesV2 はバージョン 2.1.0 を使用するようになりました。この更新により、精度が向上しました。また、新しいエンティティ、属性、関係、特性、新しいカテゴリも追加されています。

2022 年 11 月 16 日

DetectEntities の API バージョ ンの更新	API オペレーション DetectEntities はバー ジョン 0.4.0 を使用するよう になりました。この更新によ り、精度が向上し、新しい特 性が検出されるようになりました。	2022 年 11 月 16 日
StartSNOMEDCTInfer enceJob および InferSNOMEDCT の制限の更 新	StartSNOMEDCTInfer enceJob および InferSNOMEDCT API オペ レーションの個々のファイル の最大サイズは 5 KB になりま した。	2022 年 9 月 22 日
テキスト処理の更新	Amazon Comprehend Medical では、すべてのテキスト分 析 API オペレーションにお いて、関係のない空白とキャ リッジリターンを正しく解析 するためのサポートが強化さ れました。	2022 年 9 月 12 日
テキスト処理の更新	Amazon Comprehend Medical では、すべてのオントロジー API オペレーションにおい て、関係のない空白とキャ リッジリターンを正しく解析 するためのサポートが強化さ れました。	2022 年 9 月 12 日
InferSNOMEDCT の API バー ジョンの更新	Amazon Comprehend Medical API オペレーション InferSNOWMEDCT は、バー ジョン 1.1.0.20220301 を使用 するようになりました。	2022 年 7 月 8 日

[InferICD10CM の API バージョンの更新](#)

Amazon Comprehend Medical API オペレーション InferICD10CM は、バージョン 1.1.0.20220401 を使用するようになりました。

2022 年 7 月 8 日

[Amazon Comprehend Medical での InferRxNorm オペレーションの API バージョンの更新](#)

Amazon Comprehend Medical は、各 RxCUI に 2022-03-07 RxNorm および RxTerms リリースを使用するようになりました。

2022 年 1 月 29 日

[Amazon Comprehend Medical のモデルの更新](#)

Amazon Comprehend Medical では、Detect Entities V2 を使用した NEGATION 特性検出のサポートが強化されました。

2022 年 4 月 25 日

[Amazon Comprehend Medical の新機能](#)

Amazon Comprehend Medical を使用して、インターフェイス VPC エンドポイントを作成することにより、仮想プライベートクラウド (VPC) とのプライベート接続を確立できるようになりました。詳細については、「[VPC エンドポイント \(PrivateLink\)](#)」を参照してください。

2021 年 6 月 13 日

[Amazon Comprehend Medical の新機能](#)

Amazon Comprehend Medical 2020 年 5 月 4 日
で、オントロジーリンクの
バッチ操作を実行できるよう
になりました。これにより、
サービスは S3 バケットに保
存されている医療テキスト内
のエンティティを検出し、そ
れらのエンティティを標準化
されたオントロジーにリンク
できます。詳細については、
[「オントロジーリンクのバ
ッチ分析」](#)を参照してくださ
い。

[Amazon Comprehend Medical の新機能](#)

Amazon Comprehend Medical 2020 年 3 月 2 日
を使用して、日付または時
刻の式を抽出し、Amazon
Comprehend Medical によっ
て現在検出されているエン
ティティに関連付けること
ができるようになりました。
詳細については、「[エンティ
ティの検出 \(バージョン2\)](#)」を
参照してください。

[新しい特徴](#)

Amazon Comprehend Medical 2019 年 12 月 16 日
は、薬剤と病状の両方を検出
し、確立された全国的なオン
トロジーである RxNorm およ
び ICD-10-CM にエンティティ
をリンクするようになりました。
詳細については、「[オン
トロジーリンク API](#)」を参照
してください。

新機能

Amazon Comprehend Medical 2019 年 9 月 24 日
で、S3 バケットに保存されている医療テキストを処理するためにバッチ操作を利用できるようになりました。また、医療テキストの確認に使用できる新しいモデルも用意されています。詳細については、「[エンティティの検出 \(バージョン2\)](#)」を参照してください。

主な新機能

Amazon Comprehend Medical 2018 年 11 月 27 日
は、構造化されていない臨床テキスト (医師の記録、退院要約、検査結果、症例記録など) に含まれた有益な情報を検出する新しいサービスです。Amazon Comprehend Medical は、自然言語処理 (NLP) モデルを使用して機械学習の最新テクノロジーを活用することにより、この大量のデータをソートし、膨大な手作業なしには取得および使用するのが難しい貴重な情報を取得します。詳細については、「[Amazon Comprehend Medical](#)」を参照してください。

Amazon Comprehend Medical の変更ログ

以下のセクションでは、Amazon Comprehend Medical サービスの更新について詳しく説明します。

InferICD10CM の API バージョンの更新

リリース日: 2025-11-03

タイプ: 自動

InferICD10CM API オペレーションでモデルバージョン を使用するようになりました。3.2.0.20251001。この更新では、2025 年 10 月 1 日に有効になった、新しくリリースされた 2026 ICD10CM コードセットのサポートが導入されました。

InferRxNorm の API バージョンの更新

リリース日: 2025-11-03

タイプ: 自動

InferRxNorm API オペレーションでモデルバージョン を使用するようになりました。3.2.0.20221107。

InferSNOMEDCT の API バージョンの更新

リリース日: 2025-11-03

タイプ: 自動

InferSNOMEDCT API オペレーションでモデルバージョン を使用するようになりました。3.2.0.20220301。

InferICD10CM の API バージョンの更新

リリース日: 2025-08-05

タイプ: 自動

InferICD10CM API オペレーションでモデルバージョン を使用するようになりました。3.1.0.20250401。この更新では、2025 年 4 月 1 日に有効になった、新しくリリースされた 2025 年 ICD10CM コードセットのサポートが導入されました。

InferRxNorm の API バージョンの更新

リリース日: 2025-08-05

タイプ: 自動

InferRxNorm API オペレーションでモデルバージョン を使用するようになりました3.1.0.20221107。

InferSNOMEDCT の API バージョンの更新

リリース日: 2025-08-05

タイプ: 自動

InferSNOMEDCT API オペレーションでモデルバージョン を使用するようになりました3.1.0.20220301。

DetectEntities の API バージョンの更新

リリース日: 2024-06-26

タイプ: 自動

DetectEntities API オペレーションでモデルバージョン を使用するようになりました3.0.0。

DetectEntitiesV2 の API バージョンの更新

リリース日: 2024-06-26

タイプ: 自動

DetectEntitiesV2 API オペレーションでモデルバージョン を使用するようになりました3.0.0。

InferRxNorm の API バージョンの更新

リリース日: 2024-06-26

タイプ: 自動

InferRxNorm API オペレーションでモデルバージョン を使用するようになりました3.0.0.20221107。

InferSNOMEDCT の API バージョンの更新

リリース日: 2024-06-26

タイプ: 自動

InferSNOMEDCT API オペレーションでモデルバージョン を使用するようになりました3.0.0.20220301。

InferICD10CM の API バージョンの更新

リリース日: 2024-06-26

タイプ: 自動

InferICD10CM API オペレーションでモデルバージョン を使用するようになりました3.0.0.20231001。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2024-02-07

タイプ: 自動

この更新では、InferICD10CM API オペレーションが更新されました。InferICD10CM API オペレーションでモデルバージョン を使用するようになりました2.7.0.20231001。この更新では、2023年10月1日に有効になった、新しくリリースされた2024 ICD10CM コードセットのサポートが導入されました。この更新により、モデル全体の精度も向上します。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2023年9月27日

タイプ: 自動

この更新では、モデルバージョン 2.6.0.20220401 を使用するように InferICD10CM API オペレーションが更新されました。この更新により、低信頼度スコアとモデルの精度が向上します。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2023年6月12日

タイプ: 自動

この更新では、次の API オペレーションが新しいモデルバージョンを使用するようになりました。

- DetectEntities がモデルバージョン 3.0.0 を使用するようになりました。

- DetectEntitiesV2 がモデルバージョン 3.0.0 を使用するようになりました。
- InferRxNorm がモデルバージョン 3.2.0.20221107 を使用するようになりました。
- InferICD10CM はモデルバージョン 2.5.0.20220401 を使用するようになりました。
- InferSNOMEDCT がモデルバージョン 3.2.0.20220301 を使用するようになりました。

この更新により、DetectEntities、InferRxNorm、および DetectEntitiesV2 の API オペレーションの薬剤カテゴリに PAST_HISTORY 特性が追加されました。

また、この更新により、DetectEntitiesV2、DetectEntities、および InferICD10CM の病状カテゴリに QUALITY 属性が追加されました。

Amazon Comprehend Medical のテキスト分析 API オペレーションの詳細については、「[テキスト分析 API オペレーション](#)」を参照してください。

Amazon Comprehend Medical のオントロジーリンク API オペレーションの詳細については、「」を参照してください[オントロジーリンク](#)。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2023 年 4 月 28 日

タイプ: 自動

この更新では、InferRxNorm API オペレーションが更新されました。InferRxNorm API オペレーションはモデルバージョン 2.1.0.20221003 を使用するようになりました。この更新により、モデルのパフォーマンスが向上しました。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2023 年 4 月 21 日

タイプ: 自動

この更新では、InferSNOMEDCT API オペレーションが更新されました。InferSNOMEDCT API オペレーションはモデルバージョン 2.5.0.20220301 を使用するようになりました。この更新により、モデルのパフォーマンスが向上しました。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2023 年 3 月 17 日

タイプ: 自動

この更新では、InferICD10CM API オペレーションが更新されました。この更新により、DX_NAME エンティティと DIRECTION 属性の検出が向上しました。また、特性や属性の検出、API 出力での頻度/信頼度スコアに影響する、マイナーなシステム処理の調整も行っています。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2023 年 3 月 15 日

タイプ: 自動

この更新では、Amazon Comprehend Medical API オペレーション InferSNOMEDCT、InferICD10CM。DetectEntities、および DetectEntitiesV2 が更新されました。これらの更新は自動的に行われ、心臓病専門分野の病状、検査、治療、処置、および関連する属性や特性の検出が向上しました。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2023 年 2 月 8 日

タイプ: 自動

この更新では、Amazon Comprehend Medical API オペレーション InferSNOMEDCT、InferICD10CM。DetectEntities、および DetectEntitiesV2 が更新されました。これらの更新は自動的に行われ、臨床テキストをスキャンする際に病状における否定の検出精度が向上します。

Amazon Comprehend Medical InferSNOMEDCT API オペレーションの変更

リリース日: 2022 年 12 月 9 日

タイプ: 自動

Amazon Comprehend Medical API オペレーション InferSNOMEDCT が更新され、バージョン 2.2.0.20220301 を使用するようになりました。この更新により、特定された属性の全セットが返されないバグが修正されました。

StartSNOMEDCTInferenceJob API オペレーションを使用して実行されるバッチ分析では、同じ API バージョン 2.2.0.20220301 が使用されます。これらの API の使用に関する詳細については、「[SNOMED CT のリンク](#)」を参照してください。

Amazon Comprehend Medical API オペレーションの変更

リリース日: 2022 年 11 月 16 日

タイプ: 自動

この更新では、Amazon Comprehend Medical API オペレーション InferSNOMEDCT、InferICD10CM。DetectEntities、DetectEntitiesV2 が更新されました。

DetectEntities

Amazon Comprehend Medical API オペレーション DetectEntities は、API バージョン 0.4.0 を使用するようになりました。この更新により、精度が向上し、新しい特性が検出されるようになりました。

カテゴリ: TEST_TREATMENT_PROCEDURE、タイプ: PROCEDURE_NAME の場合、以下の特性が追加されました。

- NEGATION
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

カテゴリ: TEST_TREATMENT_PROCEDURE、タイプ: TEST_NAME の場合、以下の特性が追加されました。

- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

カテゴリ: TEST_TREATMENT_PROCEDURE、タイプ: TREATMENT_NAME の場合、以下の特性が追加されました。

- 否定
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

カテゴリ: MEDICAL_CONDITION、タイプ: DX_NAME の場合、以下の特性が追加されました。

- HYPOTHETICAL
- LOW_CONFIDENCE
- PERTAINS_TO_FAMILY

DetectEntitiesV2

Amazon Comprehend Medical API オペレーション DetectEntitiesV2 は、API バージョン 2.1.0 を使用するようになりました。この更新には、既存のカテゴリと特性の更新に加えて、新しいカテゴリ BEHAVIORAL_ENVIRONMENTAL_SOCIAL が含まれます。

新しいカテゴリ: BEHAVIORAL_ENVIRONMENTAL_SOCIAL には、次のタイプが追加されました。GENDER、RACE_ETHNICITY、ALLERGIES、TOBACCO_USE、ALCOHOL_CONSUMPTION、REC_DRUG

タイプ: GENDER の場合

- 属性や特性はありません。

タイプ: RACE_ETHNICITY の場合

- 属性や特性はありません。

タイプ: ALLERGIES の場合、以下の特性が追加されました。

- NEGATION
- PAST_HISTORY

タイプ: TOBACCO_USE の場合、以下の属性と特性が追加されました。

- 属性
 - AMOUNT
 - DURATION
 - FREQUENCY
- トレイト
 - NEGATION
 - PAST_HISTORY

タイプ: ALCOHOL_CONSUMPTION の場合、以下の属性と特性が追加されました。

- 属性
 - AMOUNT
 - DURATION

- FREQUENCY
- トレイト
- NEGATION
- PAST_HISTORY

タイプ: REC_DRUG_USE の場合、以下の属性と特性が追加されました。

- 属性
 - AMOUNT
 - DURATION
 - FREQUENCY
- トレイト
 - NEGATION
 - PAST_HISTORY

タイプ: UnmappedAttributes の場合、以下の属性が追加されました。

- AMOUNT

カテゴリ: MEDICAL_CONDITION、タイプ: DX_NAME の場合、以下の特性が追加されました。

- PERTAINS_TO_FAMILY
- HYPOTHETICAL
- LOW_CONFIDENCE

カテゴリ: TEST_TREATMENT_PROCEDURE、タイプ: PROCEDURE_NAME の場合、以下の特性が追加されました。

- NEGATION
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

カテゴリ: TEST_TREATMENT_PROCEDURE、タイプ: TEST_NAME の場合、以下の特性が追加されました。

- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

カテゴリ: TEST_TREATMENT_PROCEDURE、タイプ: TREATMENT_NAME の場合、以下の特性が追加されました。

- NEGATION
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

StartEntitiesDetectionV2Job API オペレーションを使用して実行されるバッチ分析では、同じ API バージョン 2.1.0 が使用されます。これらの API の使用に関する詳細については、「[エンティティの検出 \(バージョン 2\)](#)」を参照してください。

InferSNOMEDCT

Amazon Comprehend Medical API オペレーション InferSNOMEDCT は、API バージョン 2.1.0.20220301 を使用するようになりました。このバージョン更新では、以下のカテゴリとタイプに新しい特性が追加されました。

カテゴリ: MEDICAL_CONDITION、タイプ: DX_NAME の場合、以下の特性が追加されました。

- PERTAINS_TO_FAMILY
- HYPOTHETICAL
- LOW_CONFIDENCE

カテゴリ: TEST_TREATMENT_PROCEDURE、タイプ: TEST_NAME の場合、以下の特性が追加されました。

- PAST_HISTORY
- FUTURE
- HYPOTHETICAL

カテゴリ: TEST_TREATMENT_PROCEDURE、タイプ: TREATMENT_NAME の場合、以下の特性が追加されました。

- NEGATION
- PAST_HISTORY
- FUTURE
- HYPOTHETICAL

バグ修正

- この更新により、特定された属性の全セットが返されないバグが修正されました。

StartSNOMEDCTInferenceJob API オペレーションを使用して実行されるバッチ分析では、同じバージョンが使用されます。これらの API の使用に関する詳細については、「[SNOMED CT のリンク](#)」を参照してください。

InferICD10CM

Amazon Comprehend Medical API オペレーション InferICD10CM は、API バージョン 2.1.0.20220401 を使用するようになりました。このバージョン更新では、以下のカテゴリとタイプに新しい特性が追加されました。

カテゴリ: MEDICAL_CONDITION、タイプ: DX_NAME の場合、以下の特性が追加されました。

- PERTAINS_TO_FAMILY
- HYPOTHETICAL
- LOW_CONFIDENCE

バグ修正

- この更新により、特定された属性の全セットが返されないバグが修正されました。

StartRxNormInferenceJob API オペレーションを使用して実行されるバッチ分析では、同じバージョンが使用されます。InferRxNorm および StartRxNormInferenceJob API の使用に関する詳細については、「[RxNorm のリンク](#)」を参照してください。

InferRxNorm

Amazon Comprehend Medical API オペレーション InferRxNorm は、API バージョン 1.3.1.20221003 を使用するようになりました。

バグ修正

- この更新により、特定された属性の全セットが返されないバグが修正されました。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。