开发人员指南

Amazon Comprehend Medical

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

Amazon Comprehend Medical: 开发人员指南

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

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

Table of Contents

什么是 Amazon Comprehend Medical？ 1
- 重要提示 1
- Amazon Comprehend Medical 使用案例 1
- Amazon Comprehend Medical 的优势 2
- HIPAA 合规性 2
- 访问 Amazon Comprehend Medical 3
- 如何开始使用 Amazon Comprehend Medical 3

工作方式 4
- 同步实体检测 4
- 异步批量分析 4
- 本体关联 5
  - 关联到 ICD-10-CM 医学状况知识库中的概念 5
  - 链接到药物 RxNorm 知识库中的概念 5
  - 关联到 SNOMED CT 医学概念知识库中的概念 5

开始使用 6
- 步骤 1：设置账户 6
  - 注册亚马逊云科技 6
  - 创建 IAM 用户 7
  - 后续步骤 7
- 第 2 步：设置 AWS CLI 8
  - 后续步骤 8
- 步骤 3：开始使用控制台 8
  - 使用控制台分析临床文本 9
- 步骤 4：开始使用 API 11
  - 使用 AWS Command Line Interface检测医疗实体 12
  - 使用 适用于 Java 的 AWS SDK检测医疗实体 14
  - 使用 AWS SDK for Python (Boto)检测医疗实体 15

VPC 端点（AWS PrivateLink） 16
- Amazon Comprehend Medical VPC 端点的注意事项 16
- 为 Amazon Comprehend Medical 创建接口 VPC 端点 16
- 为 Amazon Comprehend Medical 创建 VPC 端点策略 17

文字分析 APIs 18
- 检测实体（版本 2） 18
  - 解剖学类别 19

行为、环境和社会健康类别 20
医学状况类别 20
药物类别 21
受保护健康信息类别 22
检查、治疗和手术类别 23
时间类别 23
检测 PHI 24
批量文本分析 APIs 28
重要提示 1
使用执行批量分析 APIs 28
使用控制台执行批量分析 29
IAM 策略 29
批量分析输出文件 31
本体论链接 APIs 33
推断 0 ICD1 CM 33
ICD-10-CM 类别 33
ICD-10-CM 类型 33
ICD-10-CM 特征 33
ICD-10-CM 属性 34
时间类别 34
类型 34
关系类型 34
输入和响应示例 35
InferRxNorm 37
重要提示 1
RxNorm 类别 38
RxNorm 类型 38
RxNorm 属性 38
RxNorm 特征 39
输入和响应示例 39
InferSNOMEDCT 42
解剖学类别 42
医学状况类别 43
检查、治疗和手术类别 43
SNOMED CT 详细信息 44
输入和响应示例 45

本体关联批量分析 52
执行批量分析 52
IAM 策略 53
批量分析输出文件 55
安全性 58
数据保护 58
身份验证和访问控制 59
身份验证 59
访问控制 60
有关管理访问的概述 61
将基于身份的策略（IAM 策略）用于 Amazon Comprehend Medical 63
Amazon Comprehend Medical API 权限参考 70
AWS 托管策略 70
使用记录亚马逊 Comprehend Medical API 调用 AWS CloudTrail 72
亚马逊 Comprehend Medical 信息在 CloudTrail 72
了解 Amazon Comprehend Medical 日志文件条目 73
合规性验证 74
恢复能力 75
基础设施安全性 75
指南和配额 76
重要提示 76
支持的区域 76
节流 76
整体配额 76
文档历史记录 82
Amazon Comprehend Medical 变更日志 88
更新了 API 版本 DetectEntities 88
更新了 DetectEntities V2 的 API 版本 89
更新了 API 版本 InferRxNorm 89
更新了 InferSNOMEDCT 的 API 版本 89
更新了 Infer ICD1 0CM 的 API 版本 89
Amazon Comprehend Medical API 操作的变更 89
Amazon Comprehend Medical API 操作的变更 90
Amazon Comprehend Medical API 操作的变更 90
Amazon Comprehend Medical API 操作的变更 90
Amazon Comprehend Medical API 操作的变更 91

Amazon Comprehend Medical API 操作的变更 91
Amazon Comprehend Medical API 操作的变更 91
Amazon Comprehend Medical API 操作的变更 91
Amazon Comprehend Medical InferSNOMEDCT API 操作的变更 92
Amazon Comprehend Medical API 操作的变更 92
..... xcvii

什么是 Amazon Comprehend Medical？

Amazon Comprehend Medical 可以检测并返回非结构化临床文本中的有用信息，例如医生记录、出院摘要、检验结果、病例记录等。Amazon Comprehend Medical 使用自然语言处理 (NLP) 模型来检测实体，这些实体是对医疗信息 [例如医学状况、药物或受保护的健康信息 (PHI)] 的文本引用。有关检测到的实体的完整列表，请参阅 检测实体（版本 2）。Amazon Comprehend Medical 还允许用户通过本体链接操作将这些检测到的实体与标准化医学知识库（ RxNorm 例如 ICD-10-CM）关联起来。

本开发人员指南中的信息适用于应用程序开发人员。本指南包括有关通过 AWS CLI 或亚马逊 Comprehend Medical 以编程方式使用亚马逊 Comprehend Medical 的信息。 APIs

Amazon Comprehend Medical 的定价与 Amazon Comprehend 的定价不同。有关更多信息，请参阅 Amazon Comprehend Medical Pricing。

支持的语言

Amazon Comprehend Medical 仅检测英语 (US-EN) 文本中的医疗实体。

重要提示

Amazon Comprehend Medical 并不代替专业的医学意见、诊断或治疗。Amazon Comprehend Medical 提供置信度分数，该分数表明对检测到的实体的准确性的信心。为您的使用案例确定正确的置信度阈值，并在需要高准确度的情况下使用高置信度阈值。对于某些使用案例，结果应由经过适当培训的人力审核人员进行审核和验证。例如，只有经过训练有素的医学专家审查准确性和进行合理的医学判断后，Amazon Comprehend Medical 才能在患者护理场景中使用。

Amazon Comprehend Medical 使用案例

您可以将 Amazon Comprehend Medical 用于以下医疗保健应用领域：

- 患者病例管理和结果：医生和医疗保健提供者可以管理和轻松访问不适合传统形式的医疗信息。与标准格式相比，病人可以用叙述的方式报告自己的健康问题，并提供更多信息。通过分析病例记录，医疗服务提供者可以在病情变得更加难以治疗、治疗费用更加昂贵之前，确定进行早期病情筛查的人选。
- 临床研究：生命科学和研究机构可以优化临床试验患者的匹配过程。通过使用 Amazon Comprehend Medical 检测临床文本中的相关信息，研究人员可以改进药物警戒，执行上市后监测以监控药物不良事件，并通过轻松检测随访记录和其他临床文本中的重要信息来评测治疗效果。例如，通过分析病人的叙述，可以更容易、更有效地监测病人对某些疗法的反应。

- 医疗账单和医疗保健收入周期管理：付款人可将分析范围扩大到临床笔记等非结构化文档。可以对诊断的更多信息进行分析和使用，以帮助从非结构化文档中确定适当的账单代码。自然语言处理 (NLP) 是计算机辅助编码 (CAC) 中最关键的组成部分。Amazon Comprehend Medical 采用最新的 NLP 技术来分析临床文本，这有助于缩短创收时间并提高报销准确性。
- 本体关联：使用本体关联功能从临床文本中检测实体，并将这些实体与常见医学本体中的标准化概念关联起来。Infer ICD1 0CM 将可能的疾病识别为实体。Infer ICD1 0CM 将这些实体与2021年版《国际疾病分类，第十修订版，临床修改》（ICD-10-CM）中的唯一代码联系起来。 InferRxNorm将临床文本中列出的药物识别为实体，并将这些实体与美国国家医学图书馆RxNorm数据库中的标准化概念标识符关联起来。InfersNomedCT 将医学概念（例如医学状况和解剖学、医学检查或治疗和手术）检测为实体，并将它们与医学系统命名法 – 临床术语 (SNOMED CT) 本体中的代码关联起来。

Amazon Comprehend Medical 的优势

使用 Amazon Comprehend Medical 的部分优势包括：

- 轻松而强大的自然语言处理集成到您的应用程序中 — 用于在应用程序中构建文本分析功能， APIs 以实现强大而准确的自然语言处理。
- 准确性：使用深度学习技术精确分析文本。我们的模型不断使用跨多个领域的新数据进行训练，以提高准确性。
- 可扩展性：检测来自多个文档的信息，从而可以快速了解患者的健康和护理。
- 与其他 AWS 服务集成：Amazon Comprehend Medical 旨在与其他 AWS 服务（例如 Amazon S3 和 AWS Lambda）无缝协作。将您的文档存储在亚马逊 S3 中，使用 Firehose 分析实时数据，或者使用 Amazon Transcribe 将患者的叙述转录为可供亚马逊 Comprehend Medical 分析的文本。Support for AWS Identity and Access Management (IAM) 使您可以轻松安全地控制对亚马逊 Comprehend Medical 运营的访问权限。借助 IAM，您可以创建和管理 AWS 用户和组，以便向开发人员和最终用户授予适当的访问权限。
- 低成本：只需为您分析的文档付费。没有最低费用或预付费用。

HIPAA 合规性

这是一项符合 HIPAA 要求的服务。有关 AWS《1996 年美国健康保险流通与责任法案》(HIPAA) 以及使用 AWS 服务处理、存储和传输受保护的健康信息 (PHI) 的更多信息，请参阅 HIPAA 概述。

必须对包含 PHI 的 Amazon Comprehend Medical 的连接进行加密。默认情况下，所有与 Amazon Comprehend Medical 的连接都使用基于 TLS 的 HTTPS。Amazon Comprehend Medical 不会持续存储客户内容。因此，您无需在服务中配置静态加密。

访问 Amazon Comprehend Medical

1. AWS 管理控制台：提供一个 Web 界面，您可以使用该界面访问 Amazon Comprehend Medical。
2. AWS 命令行界面（AWS CLI）：提供用于众多 AWS 服务（包括 Amazon Comprehend Medical）的命令，并且在 Windows、macOS 和 Linux 上受支持。有关安装 AWS CLI 的更多信息，请参阅 AWS 命令行界面。
3. AWS SDKs — AWS 提供 SDKs（软件开发套件），其中包括适用于各种编程语言和平台（Java、Python、Ruby、.NET、iOS、Android 等）的库和示例代码。它们 SDKs 提供了一种便捷的方式来创建对亚马逊 Comprehend Medical 和 AWS 的编程访问权限。有关更多信息，请参阅 AWS SDKs。

如何开始使用 Amazon Comprehend Medical

如果您是首次接触 Amazon Comprehend Medical 的用户，我们建议您按顺序阅读以下章节：

1. Amazon Comprehend Medical 工作原理：本节介绍 Amazon Comprehend Medical 的概念。
2. Amazon Comprehend Medical 入门：本节介绍如何设置账户和测试 Amazon Comprehend Medical。

Amazon Comprehend Medical 工作原理

Amazon Comprehend Medical 使用预训练的自然语言处理 (NLP) 模型，通过实体检测来分析非结构化临床文本。实体是指对医疗信息的文本引用，例如医学状况、药物或受保护的健康信息 (PHI)。有些操作可以更进一步，检测实体，然后将这些实体与标准化本体关联起来。该模型在大量医学文本上持续训练，因此您无需提供训练数据。所有结果都包括置信度分数，该分数表明 Amazon Comprehend Medical 对检测到的实体的准确性的信心。

实体检测和本体关联既可以作为同步操作执行，也可以作为异步操作执行：

- 同步操作：支持对单个文档进行分析，从而将分析结果直接返回到您的应用程序。当您创建一次处理一个文档的交互式应用程序时，请使用单文档操作。
- 异步操作：支持对存储在 Amazon S3 桶中的文档集合或批次进行分析。分析结果将返回在 S3 桶中。

> **Note**
>
> Amazon Comprehend Medical 只能分析英语 (US-EN) 文本。

同步实体检测

DetectEntitiesV2 和 D etectPhi 操作可从单个文档中检测非结构化临床文本中的实体。您向 Amazon Comprehend Medical 服务发送文档，并在回复中收到分析结果。

异步批量分析

StartEntitiesDetectionV2Job 和 Start Jo PHIDetectionb 操作会启动异步作业，以检测对医疗信息（例如医疗状况、治疗、检查和结果）的引用，或者存储在 Amazon S3 存储桶中的受保护健康信息。检测作业的输出将写入单独的 Amazon S3 桶，用于进一步处理或下游分析。

Start ICD1 0 CMInference J ob 和StartRxNormInferenceJob操作启动本体论，将检测实体的批处理操作链接起来，并将这些实体链接到 RxNorm 和 ICD-10-CM 知识库中的标准化代码。

本体关联

Infer ICD1 0CM、InfersnomedCT 和InferRxNorm手术可检测潜在的疾病和药物，并将它们分别与 ICD-10-CM、SNOMED CT 或知识库中的代码关联起来。 RxNorm 您可以使用本体关联批量分析来分析一组文档或单个大型文档。通过使用控制台或本体链接批处理 APIs，您可以执行启动、停止、列出和描述正在进行的批量分析作业的操作。

关联到 ICD-10-CM 医学状况知识库中的概念

Infer ICD1 0CM 操作可检测潜在的疾病，并将其与2019年版《国际疾病分类，第10修订版，临床修改》（ICD-10-CM）中的代码关联起来。对于检测到的每种潜在医学状况，Amazon Comprehend Medical 都会列出匹配的 ICD-10-CM 代码和描述。结果中列出的医学状况包括置信度分数，该分数表明 Amazon Comprehend Medical 对实体与结果中的概念匹配准确性的信心程度。

链接到药物 RxNorm 知识库中的概念

该InferRxNorm手术将患者记录中列出的药物识别为实体。它将实体链接到国家医学图书馆 RxNorm 数据库中的概念标识符 (rxCUI)。每个 RxCUI 都是独一无二的，具有不同的强度和剂量形式。结果中列出的药物包括置信度分数，这表明Amazon Comprehend Medical对与知识库中的概念相匹配的实体的准确性有信心。 RxNormAmazon Comprehend Medical 根据置信CUIs 度分数按降序列出了可能与其检测到的每种药物相匹配的顶级处方药物。

关联到 SNOMED CT 医学概念知识库中的概念

InferSNOMEDCT 操作将可能的医学概念识别为实体，并将其与 2021 年 3 月版本的《医学系统术语表 (SNOMED CT)》中的代码关联起来。SNOMED CT 提供了全面的医学概念词汇，包括医学状况和解剖学，以及医学检查、治疗和手术。对于每个匹配的概念 ID，Amazon Comprehend Medical 会返回排名前五的医学概念，每个概念都有置信度分数和情境信息，例如相关特征和属性。然后，当与 SNOMED CT 多层次结构一起使用时，SNOMED CT 概念 IDs 可用于构建患者临床数据，用于医学编码、报告或临床分析。

Amazon Comprehend Medical 入门

要开始使用 Amazon Comprehend Medical，请设置一个 AWS 账户并创建 AWS Identity and Access Management 一个 (IAM) 用户。要使用 Amazon Comprehend Medical CLI，请下载并对其进行配置。

主题

- 步骤 1：设置 AWS 账户并创建管理员用户
- 步骤 2：设置 AWS Command Line Interface (AWS CLI)
- 第 3 步：开始使用 Amazon Comprehend Medical 控制台
- 第 4 步：开始使用亚马逊 Comprehend Medical APIs

步骤 1：设置 AWS 账户并创建管理员用户

首次使用 Amazon Comprehend Medical 前，请完成以下任务：

1. 注册亚马逊云科技
2. 创建 IAM 用户

注册亚马逊云科技

当您注册 Amazon Web Services (AWS) 时，您的 AWS 账户会自动注册所有 AWS 服务，包括 Amazon Comprehend Medical。您只需为使用的服务付费。

借助 Amazon Comprehend Medical，您仅需为实际使用的资源付费。如果您是 AWS 新客户，还可以免费试用 Amazon Comprehend Medical。有关更多信息，请参阅 AWS 免费使用套餐。

如果您已有 AWS 账户，请跳至下一节。

创建 AWS 账户

1. 打开https://portal.aws.amazon.com/billing/注册。
2. 按照屏幕上的说明操作。

   在注册时，将接到电话或收到短信，要求使用电话键盘输入一个验证码。

当您注册时 AWS 账户，就会创建AWS 账户根用户一个。根用户有权访问该账户中的所有 AWS 服务 和资源。作为最佳安全实践，请为用户分配管理访问权限，并且只使用根用户来执行需要根用户访问权限的任务。

记录您的 AWS 账户 ID，因为进行下一个任务时需要用到。

创建 IAM 用户

AWS 中的服务（如 Amazon Comprehend Medical）要求您在访问时提供凭证。这允许服务确定您是否有权访问服务的资源。

我们强烈建议您使用 AWS Identity and Access Management (IAM) 访问 AWS，而不是使用 AWS 账户的证书。要使用 IAM 访问 AWS，请创建 IAM 用户，将该用户添加到具有管理权限的 IAM 组，然后向该 IAM 用户授予管理权限。您随后可以使用一个专用 URL 和该 IAM 用户的凭证访问 AWS。

本指南中的入门练习假定您拥有具有管理员权限的用户，即 `adminuser`。

创建管理员和登录控制台

1. 在 AWS 账户中创建名为 `adminuser` 的用户。有关说明，请参考《IAM 用户指南》中的创建您的第一个 IAM 用户和管理员组。
2. AWS Management Console 使用特殊的 URL 登录。有关更多信息，请参阅《IAM 用户指南》中的用户如何登录您的账户。

有关 IAM 的更多信息，请参阅以下文档：

- AWS Identity and Access Management (IAM)
- 入门
- IAM 用户指南

后续步骤

步骤 2：设置 AWS Command Line Interface (AWS CLI)

步骤 2：设置 AWS Command Line Interface (AWS CLI)

您无需使用 AWS CLI 即可执行入门练习中的步骤。但是，本指南中的某些其他练习会需要它。如果您愿意，可以跳过此步骤并转至第 3 步：开始使用 Amazon Comprehend Medical 控制台，然后 AWS CLI 再进行设置。

要设置 AWS CLI

1. 下载并配置 AWS CLI。有关说明，请参阅《AWS Command Line Interface 用户指南》中的以下主题：
   - 开始设置 AWS Command Line Interface
   - 配置 AWS Command Line Interface
2. 在 AWS CLI 配置文件中，为管理员添加一个命名的配置文件：

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

   执行 AWS CLI 命令时使用此配置文件。有关命名配置文件的更多信息，请参阅《AWS Command Line Interface 用户指南》中的命名配置文件。有关 AWS 区域的列表，请参阅 https://docs.aws.amazon.com/general/latest/gr/rande.html 中的Amazon Web Services 一般参考区域和终端节点。
3. 在命令提示符处键入以下帮助命令来验证设置：

   ```
   aws help
   ```

后续步骤

第 3 步：开始使用 Amazon Comprehend Medical 控制台

第 3 步：开始使用 Amazon Comprehend Medical 控制台

开始使用 Comprehend Medical 控制台的最简单方法是分析一个简短的文本文件。如果您尚未查看Amazon Comprehend Medical 工作原理中的概念和术语，我们建议您在继续之前先去查看一下。

主题

- 使用控制台分析临床文本

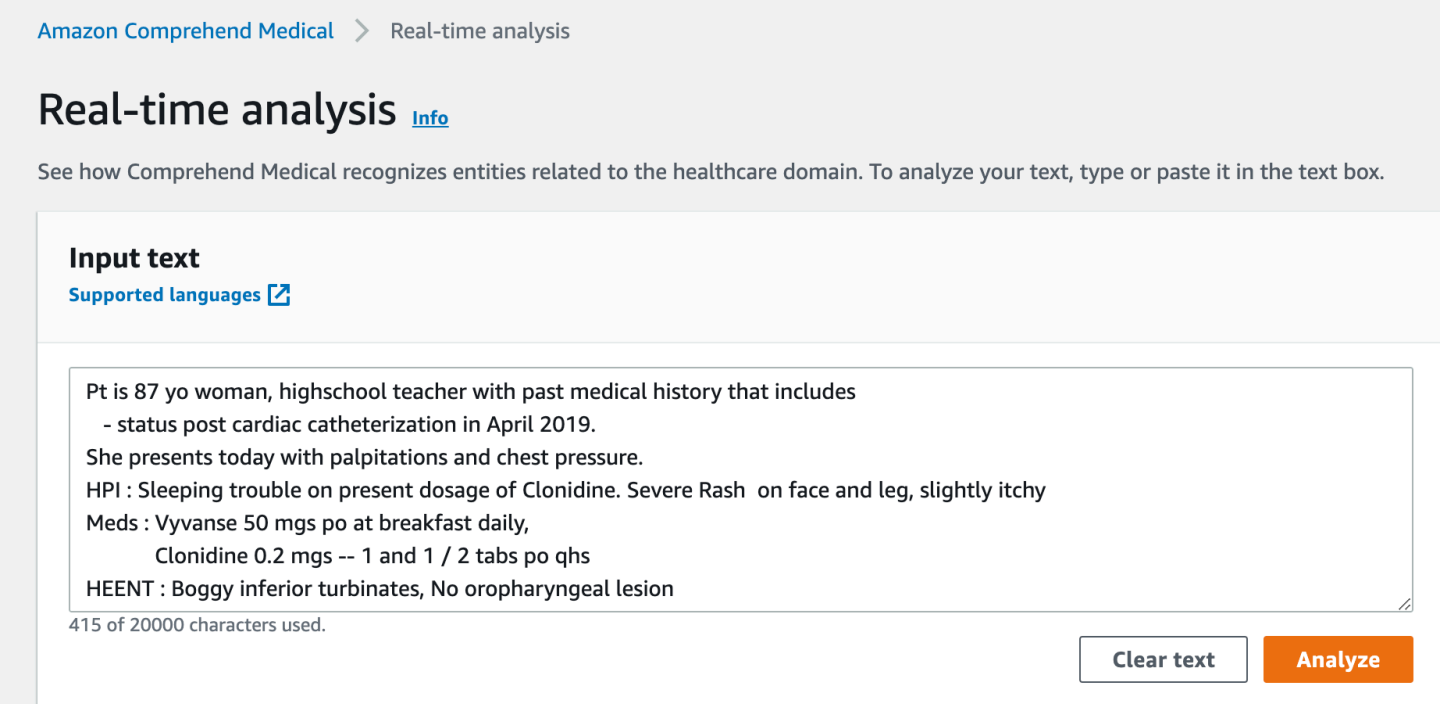
使用控制台分析临床文本

Comprehend Medical 控制台允许您分析最长 20,000 个字符的临床文本内容。结果显示在控制台中，以便您可以查看分析。

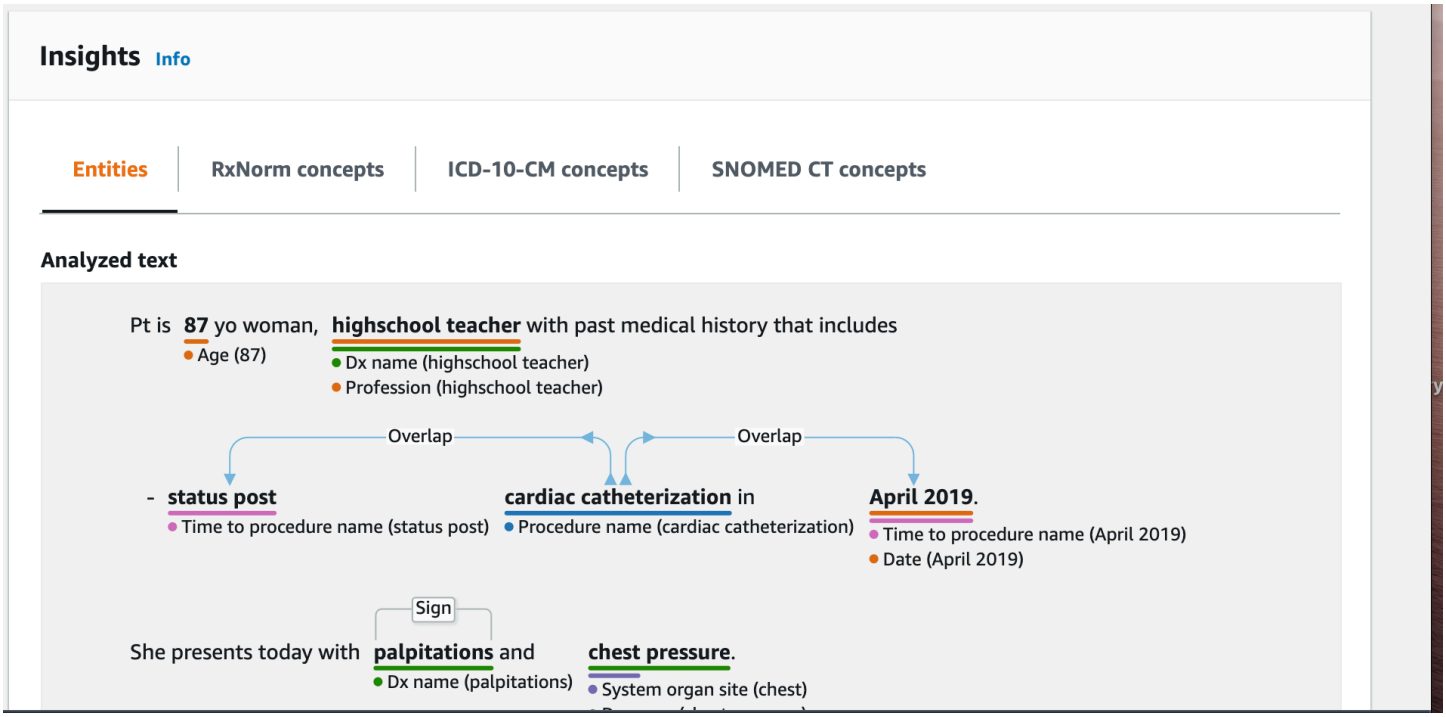
要开始分析文档，请登录 AWS Management Console 并打开 Comprehend Medical 控制台。

在 Comprehend Medical 下，选择实时分析。

控制台显示示例文本和对该文本的分析：



您可以用自己的英文文本替换示例文本，然后选择分析来分析您的文本。



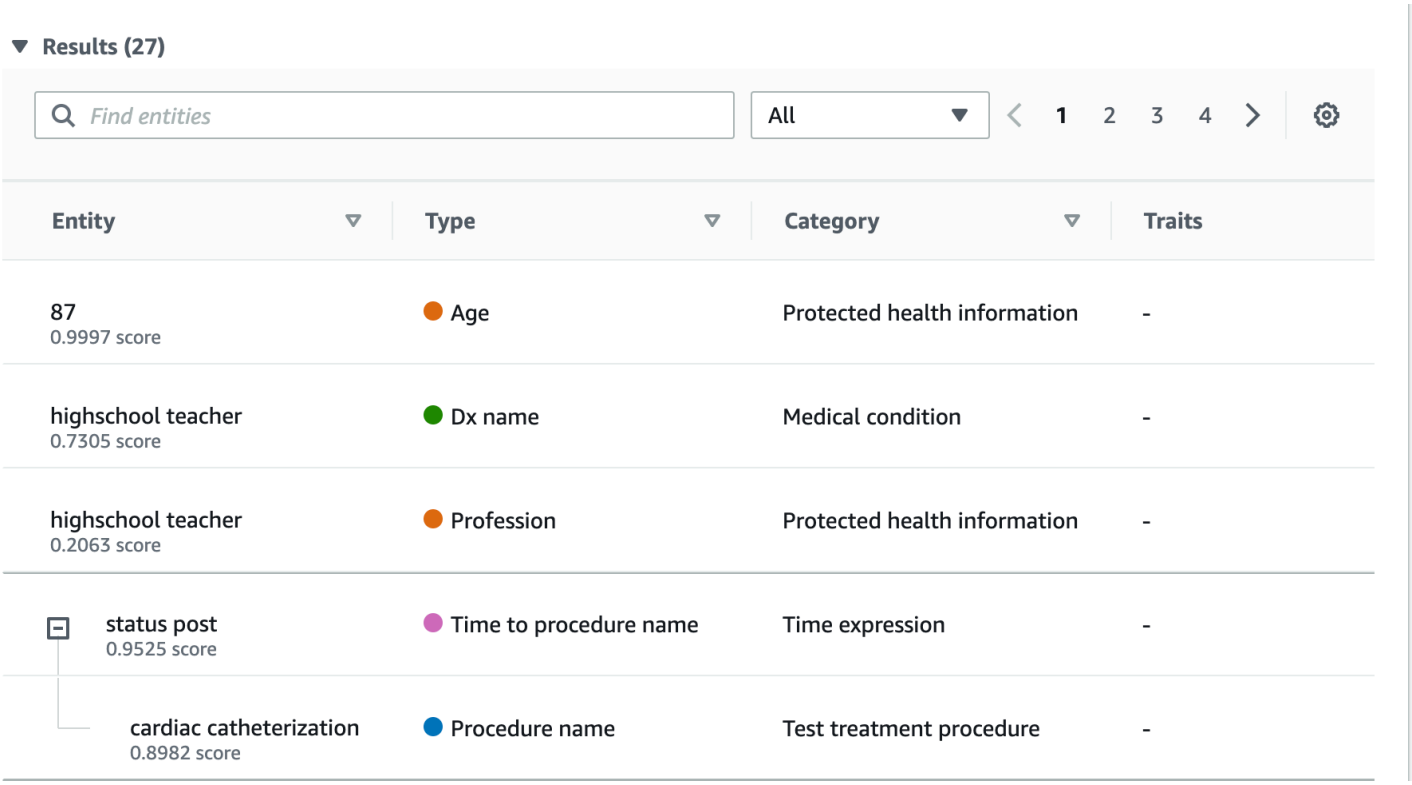
在输入文本下方，已分析的文本进行了颜色编码，以指示实体类别：

- 橙色标签用于标识 PHI 数据。
- 红色标签用于标识药物。
- 绿色标签用于标识医学状况。
- 蓝色标签用于标识检查、治疗或手术 (TTP)。
- 紫色标签用于标识解剖学。
- 粉色标签用于标识时间表达式。

有关更多信息，请参阅 Amazon Comprehend Medical 工作原理。

在控制台的输入框下方，已分析文本窗格显示有关文本的更多信息。

实体部分显示文本中找到的实体的卡片：



每张卡片都显示文本及其实体类型。

在每个实体旁边，分数表示 Comprehend Medical 对将文本识别为所示实体类型的信心。

要查看请求和结果的 JSON 结构，请选择应用程序集成。JSON 结构与操作返回的结构相同。

下一个步骤

第 4 步：开始使用亚马逊 Comprehend Medical APIs

第 4 步：开始使用亚马逊 Comprehend Medical APIs

以下示例演示了如何使用、Java 和 Python 来使用 Amazon Comprehend Medical 操作。 AWS CLI使用它们了解 Amazon Comprehend Medical 操作，并将其作为您自己的应用程序的构建块。

要运行 AWS CLI 和 Python 示例，请安装 AWS CLI。有关更多信息，请参阅 步骤 2：设置 AWS Command Line Interface (AWS CLI)。

要运行 Java 示例，请安装 适用于 Java 的 AWS SDK。有关安装 适用于 Java 的 AWS SDK的说明，请参阅 Set up the AWS SDK for Java。

主题

- 使用 AWS Command Line Interface检测医疗实体
- 使用 适用于 Java 的 AWS SDK检测医疗实体
- 使用 AWS SDK for Python (Boto)检测医疗实体

使用 AWS Command Line Interface检测医疗实体

以下示例演示如何使用`DetectEntitiesV2`操作返回文本中检测到的医疗实体。 AWS CLI 要运行示例，您必须安装 AWS CLI。有关更多信息，请参阅 the section called “第 2 步：设置 AWS CLI”。

此示例的格式适用于 Unix、Linux 和 macOS。对于 Windows，请将每行末尾的反斜杠 (\) Unix 行继续符替换为脱字号 (^)。

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

响应将为以下内容：

```
{
    "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": []
```

```
}
```

使用 适用于 Java 的 AWS SDK检测医疗实体

以下示例使用 Java 的 `DetectEntitiesV2` 操作。要运行示例，请安装 适用于 Java 的 AWS SDK。有关安装的说明 适用于 Java 的 AWS SDK，请参阅设置适用于 Java 的 AWS 开发工具包。

```
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);
    }
}
```

输出包含在输入文本中找到的三个实体，以及它们在输入文本中的位置。每个实体还列出了正确识别该实体的可信度。以下输出显示了前面示例中的 Generic_Name、Dosage 和 Frequency 实体。

```
{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: []}]}
```

使用 AWS SDK for Python (Boto)检测医疗实体

以下示例使用 Python 的 DetectEntitiesV2 操作。要运行示例，请安装 AWS CLI。有关更多信息，请参阅 the section called “第 2 步：设置 AWS CLI”。

```
import boto3
client = boto3.client(service_name='comprehendmedical', region_name='YOUR REGION')
result = client.detect_entities(Text= 'cerealx 84 mg daily')
entities = result['Entities']
for entity in entities:
    print('Entity', entity)
```

输出包含在输入文本中找到的三个实体，以及它们在输入文本中的位置。每个实体还列出了正确识别该实体的可信度。以下输出显示了前面示例中的 Generic_Name、Dosage 和 Frequency 实体。

```
('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 和接口 VPC 端点 (AWS PrivateLink)

您可以通过创建接口 VPC 端点在 VPC 和 Amazon Comprehend Medical 之间建立私有连接。接口 VPC 终端节点由这项技术提供支持，无需互联网网关AWS PrivateLink、NAT 设备、VPN 连接或连接即可使用该技术私密访问 Amazon Comprehend M APIs edical。 AWS Direct Connect 您的 VPC 中的实例不需要公有 IP 地址即可与 Amazon Comprehend Medical 通信。 APIsVPC 和 Amazon Comprehend Medical 之间的流量不会脱离 Amazon 网络。

每个接口端点均由子网中的一个或多个弹性网络接口表示。

有关更多信息，请参阅 A mazon VPC 用户指南中的接口 VPC 终端节点 (AWS PrivateLink)。

Amazon Comprehend Medical VPC 端点的注意事项

在为 Amazon Comprehend Medical 设置接口 VPC 端点之前，请务必查看《Amazon VPC 用户指南》中的接口端点属性和限制。

Amazon Comprehend Medical 支持从 VPC 调用它的所有 API 操作。

为 Amazon Comprehend Medical 创建接口 VPC 端点

您可以使用亚马逊 VPC 控制台或 () 为 Amazon Comprehend Medical 服务创建 VPC 终端节点。 AWS Command Line Interface AWS CLI有关更多信息，请参阅《Amazon VPC User Guide》中的 Creating an interface endpoint。

使用以下服务名称为 Amazon Comprehend Medical 创建 VPC 端点：

- com.amazonaws。 *region*.comprehendmedical

如果为端点启用私有 DNS，则可以使用其默认 DNS 名称作为区域，向 Amazon Comprehend Medical 发送 API 请求。例如，`comprehendmedical.us-east-1.amazonaws.com`。

有关更多信息，请参阅《Amazon VPC 用户指南》中的通过接口端点访问服务。

为 Amazon Comprehend Medical 创建 VPC 端点策略

您可以为 VPC 端点附加控制对 Amazon Comprehend Medical 的访问的端点策略。该策略指定以下信息：

- 可执行操作的主体。
- 可执行的操作。
- 可对其执行操作的资源。

有关更多信息，请参阅《Amazon VPC 用户指南》中的使用 VPC 端点控制对服务的访问。

示例：Amazon Comprehend Medical 操作的 VPC 端点策略

下面是用于 Amazon Comprehend Medical 的端点策略示例。当附加到端点时，此策略会向所有资源上的所有主体授予对 Amazon Comprehend Medical `DetectEntitiesV2` 操作的访问权限。

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

文本分析 API 操作

使用 Amazon Comprehend Medical 检查临床文档，并使用预先训练的自然语言处理 (NLP) 模型获得有关其内容的各种见解。您可以对单个文件执行分析，也可以对存储在 Amazon Simple Storage Service (S3) 存储桶中的多个文件执行批量分析。

借助 Amazon Comprehend Medical，您可以对文档执行以下操作：

- 检测实体（版本 2）：检查非结构化临床文本，检测文本中对医疗信息的引用，如医学状况、治疗、检查和结果以及药物。此版本使用的模型与最初的 Detect entities API 不同，并且输出中有一些更改。
- 检测 PHI：检查非结构化临床文本，以检测对受保护的健康信息 (PHI) 的文本引用，例如姓名和地址。

Amazon Comprehend Medical 还包括多个 API 操作，您可以使用这些操作对临床文件进行批量文本分析。要了解有关使用这些 API 操作的更多信息，请参阅 the section called “批量文本分析 APIs”。

主题

- 检测实体（版本 2）
- 检测 PHI
- 批量文本分析 APIs

检测实体（版本 2）

使用 DetectEntitiesV2 检测单个文件中的实体，或者使用 StartEntitiesDetectionV2J ob 对多个文件进行批量分析。您可以检测以下类别的实体：

- `ANATOMY`：检测对身体部位或身体系统以及这些部位或系统的位置的表述。
- `BEHAVIORAL_ENVIRONMENTAL_SOCIAL`：检测环境中影响个人健康的行为和状况。其中包括吸烟、饮酒、吸毒、过敏、性别和种族/民族。
- `MEDICAL_CONDITION`：检测医学状况的体征、症状和诊断。
- `MEDICATION`：检测患者的药物和剂量信息。
- `PROTECTED_HEALTH_INFORMATION`：检测患者的个人信息。
- `TEST_TREATMENT_PROCEDURE`：检测用于确定医学状况的程序。

- TIME_EXPRESSION：检测与检测到的实体关联的时间实体。

DetectEntitiesV2 操作可以检测到所有六个类别。要进行特定于检测 PHI 的分析，请对单个文件使用 detectPhi，使用 Start PHIDetectionJob 进行批量分析。

Amazon Comprehend Medical 可以检测以下类别的信息：

- 实体：对人员、治疗方法、药物和医学状况等相关对象名称的文字表述。例如，ibuprofen。
- 类别：实体所属的广义分类。例如，布洛芬属于 MEDICATION 类别。
- 类型：检测到的实体在单个类别中的类型。例如，布洛芬属于 MEDICATION 类别的 GENERIC_NAME 类型。
- 属性：与实体相关的信息，例如药物剂量。例如，200 mg 是布洛芬实体的属性。
- 特征：Amazon Comprehend Medical 根据背景信息对实体的了解。例如，如果患者未服用某种药物，则该药物具有 NEGATION 特征。
- 关系类型：实体与属性之间的关系。

Amazon Comprehend Medical 会为您提供实体在输入文本中的位置。在 Amazon Comprehend 控制台中，位置以图形方式显示。当您使用 API 时，它会通过数值偏移来显示位置。

每个实体和属性都包含一个得分，用来表明 Amazon Comprehend Medical 对检测准确性的置信度。每个属性也有一个关系分数。该得分可以表明 Amazon Comprehend Medical 对该属性与其父实体之间关系的准确性的置信度。请针对您的使用场景确定适当的置信度阈值。需要高准确性时，请使用高置信度阈值。筛选出不符合阈值的数据。

解剖学类别

ANATOMY 类别可以检测对身体部位或身体系统以及这些部位或系统的位置的表述。

类型

- SYSTEM_ORGAN_SITE：身体系统、解剖学位置或区域以及身体部位。

Attributes

- DIRECTION：方位词。例如左、右、内侧、外侧、上、下、后、前、远端、近端、对侧、双侧、同侧、背侧、腹侧等。

行为、环境和社会健康类别

BEHAVIORAL_ENVIRONMENTAL_SOCIAL 类别可以检测对环境中影响个人健康的行为和状况的表述。

类型

- ALCOHOL_CONSUMPTION：用使用状态、频率、数量和持续时间来说明患者的饮酒量。
- ALLERGIES：定义患者的过敏和对过敏原的反应。
- GENDER：表明性别认同。
- RACE_ETHNICITY: 表明患者所属的特定种族和族裔群体。
- REC_DRUG_USE：用使用状态、频率、数量和持续时间来说明患者的吸毒量。
- TOBACCO_USE：用使用状态、频率、数量和持续时间来说明患者的吸烟量。

Attributes

检测到的以下属性仅适用于 ALCOHOL_CONSUMPTION、TOBACCO_USE 和 REC_DRUG_USE：

- AMOUNT：酒精、烟草或毒品的使用量。
- DURATION：酒精、烟草或毒品的使用时间。
- FREQUENCY：酒精、烟草或毒品的使用频率。

特征

检测到的以下特征仅适用于 ALCOHOL_CONSUMPTION、ALLERGIES、TOBACCO_USE 和 REC_DRUG_USE：

- NEGATION：表明结果为阴性或行动未执行。
- PAST_HISTORY：表明酒精、烟草或毒品的使用来自患者的过去（现在之前）。

医学状况类别

MEDICAL_CONDITION 类别可以检测医学状况的体征、症状和诊断。该类别具有一种实体类型、四个属性和四个特征。一个或多个特征可以与一个类型关联。有关属性的背景信息及其与诊断的关系会通过 RELATIONSHIP_EXTRACTION. 被检测到并映射到 DX_NAME。例如，从文本“左腿慢性

疼痛”中，“慢性”会被检测为属性 ACUITY，“左”会被检测为属性 DIRECTION，“腿”会被检测为属性 SYSTEM_ORGAN_SITE。这些属性的关系会被映射到医学状况实体“疼痛”，并附带一个置信度得分。

类型

- DX_NAME：列出的所有医学状况。DX_NAME 类型包括目前的疾病、就诊原因和病史。

Attributes

- ACUITY：确定疾病的猛烈程度，例如慢性、急性、突发性、持续性或渐进性。
- DIRECTION：方位词。例如左、右、内侧、外侧、上、下、后、前、远端、近端、对侧、双侧、同侧、背侧或腹侧。
- SYSTEM_ORGAN_SITE：解剖学位置。
- QUALITY：医学状况的任何描述词语，例如阶段或等级。

特征

- DIAGNOSIS：被确定为症状的原因或结果的医学状况。症状可以通过体检结果、实验室或放射学报告或任何其他方式发现。
- HYPOTHETICAL：表明医学状况是一种假设。
- LOW_CONFIDENCE：表明医学状况具有高度不确定性。这与提供的置信度分数没有直接关系。
- NEGATION：表明结果为阴性或行动未执行。
- PERTAINS_TO_FAMILY：表明医学状况与患者家属有关，与患者无关。
- SIGN：医生报告的医学状况。
- SYMPTOM：患者报告的医学状况。

药物类别

MEDICATION 类别可以检测患者的药物和剂量信息。一个或多个属性可以应用于一个类型。

类型

- BRAND_NAME：药物或治疗剂的受版权保护的品牌名称。
- GENERIC_NAME：药物或治疗剂的非品牌名称、成分名称或配方混合物。

Attributes

- DOSAGE：医嘱用药量。
- DURATION：用药持续时间。
- FORM：药物的形式。
- FREQUENCY：用药频率。
- RATE: 药物的给药率（主要用于药物输液或 IVs）。
- ROUTE_OR_MODE：药物的用药方法。
- STRENGTH：药物强度。

特征

- NEGATION：表明患者没有用药。
- PAST_HISTORY：表明检测到的药物来自患者的过去（现在之前）。

受保护健康信息类别

PROTECTED_HEALTH_INFORMATION 类别可以检测患者的个人信息。要详细了解这一操作，请参阅检测 PHI 。

类型

- ADDRESS：任何机构、单位或机构内的病房的地址的所有地理细分信息。
- AGE：提及的所有年龄信息、年龄范围或任何年龄。其中包括患者、家庭成员或其他人的信息。除非另有说明，否则默认单位为岁。
- EMAIL：任何电子邮件地址。
- ID：社会保障号码、病历号、机构识别号、临床试验号、证书或执照号、车辆或设备号、护理地点或提供者。其中还包括患者的任何生物识别数字，例如身高、体重或实验室数值。
- NAME：所有姓名。通常是患者、家属或提供者的姓名。
- PHONE_OR_FAX：任何电话、传真或寻呼机号码。不包括指定电话号码，例如 1-800-QUIT-NOW 和 911。
- PROFESSION：与患者或患者家属有关的任何职业或雇主。其中不包括记录中提及的临床医生的职业。

检查、治疗和手术类别

TEST_TREATMENT_PROCEDURE 类别可以检测用于确定医学状况的程序。一个或多个属性可以与 TEST_NAME 类型的实体相关。

类型

- PROCEDURE_NAME：对患者进行的一次性干预措施，用于治疗疾病或为患者提供护理。
- TEST_NAME：对患者执行的诊断、测量、筛查或评级程序，最后可能产生结果数值。其中包括用于确定诊断、排除或确定病情或对患者进行评级或打分的任何程序、流程、评估或评分。
- TREATMENT_NAME：在一段时间内为对抗疾病或失调而采取的干预措施。其中包括药物，例如抗病毒药物和疫苗接种。

Attributes

- TEST_VALUE：检查结果。仅适用于 TEST_NAME 实体类型。
- TEST_UNIT：检查结果数值可能采用的计量单位。仅适用于 TEST_NAME 实体类型。

特征

- FUTURE：表明检查、治疗或手术是将在临床记录中的主题事件之后发生的行动或事件。
- HYPOTHETICAL：表明检查、治疗或手术是一种假设。
- NEGATION：表明结果为阴性或行动未执行。
- PAST_HISTORY：表明检查、治疗或手术来自患者的过去（现在之前）。

时间类别

TIME_EXPRESSION 类别可以检测与时间相关的实体。其中包括日期和时间等实体，例如 “三天前”、“今天”、“当前”、“入院当天”、“上个月”或“16 天”。此类别中的结果仅在与实体关联时才会返回。例如，“患者昨天服用了 200 毫克布洛芬”会将 Yesterday 作为与 GENERIC_NAME 实体“布洛芬”重叠的 TIME_EXPRESSION 实体返回。但是，在“患者昨天遛过狗”中，“昨天”不会被认定为一个实体。

类型

- TIME_TO_MEDICATION_NAME：服药的日期。此类型的特定属性为 BRAND_NAME 和 GENERIC_NAME。

- TIME_TO_DX_NAME：医学状况的发生日期。此类型的属性为 DX_NAME。
- TIME_TO_TEST_NAME：接受检查的日期。此类型的属性为 TEST_NAME。
- TIME_TO_PROCEDURE_NAME：实施手术的日期。此类型的属性为 PROCEDURE_NAME。
- TIME_TO_TREATMENT_NAME：进行治疗的日期。此类型的属性为 TREATMENT_NAME。

关系类型

- 实体与属性之间的关系。识别出的 Relationship_type 如下：

  Overlap — TIME_EXPRESSION 与检测到的实体一致。

检测 PHI

如果在扫描临床文本时只想检测受保护健康信息 (PHI) 数据，请使用 DetectPHI 操作。要检测临床文本中的所有可用实体，请使用 DetectEntitiesV2。

本 API 最适合只需要检测 PHI 实体的使用场景。要了解非 PHI 类别的信息，请参阅检测实体（版本 2）。

> **Important**
>
> Amazon Comprehend Medical 提供置信度分数，该分数表明对检测到的实体的准确性的信心。请评估这些置信度分数，为您的使用场景确定正确的置信度阈值。对于特定的合规使用场景，我们建议您通过额外的人工审查或其他方法来确认检测到的 PHI 的准确性。

根据 HIPAA 法案，我们必须特别谨慎对待与 18 个标识符对应的 PHI。Amazon Comprehend Medical 会检测与这些标识符关联的实体，但这些实体并不会与 Safe Harbor 方法指定的列表一一对应。并非所有标识符都包含在非结构化临床文本中，但是 Amazon Comprehend Medical 能够检测所有相关的标识符。这些标识符由可用于识别患者身份的数据组成，显示在以下列表中。有关更多信息，请参阅美国政府卫生与公共服务网站上的健康信息隐私。

每个 PHI 相关实体都包含一个分数（响应中的 Score），用来表明 Amazon Comprehend Medical 对检测准确性的置信度。请为您的使用场景确定正确的置信度阈值，并筛选出不符合阈值的实体。在识别 PHI 时，最好使用低置信度阈值进行筛选，以便检测到更多实体。在合规使用场景中不使用检测到的实体的值时，这一点尤其重要。

通过运行 detectPhi 或 DetectEntities V2 操作可以检测到以下与 PHI 相关的实体：

检测到的 PHI 实体

实体	描述	HIPAA 类别
AGE	临床记录中提及的患者、家属或其他人的所有年龄信息、年龄范围或任何年龄。除非另有说明，否则默认单位为岁。	3. 与个人相关的日期
DATE	与患者或患者护理相关的任何日期。	3. 与个人相关的日期
NAME	临床记录中提及的所有姓名，通常属于患者、家属或提供者。	1. 姓名
PHONE_OR_FAX	任何电话、传真、寻呼机号码；不包括指定电话号码，例如 1-800-QUIT-NOW 和 911。	4. 电话号码 5. 传真号码
EMAIL	任何电子邮件地址。	6. 电子邮件地址
ID	与患者身份相关的任何数字。其中包括其社会保障号码、病历号、机构识别号、临床试验号、证书或执照号、车辆或设备号。还包括生物识别数字以及识别护理地点或提供者的数字。	7. 社会保障号码 8. 病历号 9. 健康计划号码 10. 账户号码 11. 证书/许可证号 12. 车辆识别号 13. 设备号 16. 生物识别信息 18。任何其他识别特征
URL	任何网址。	14。 URLs

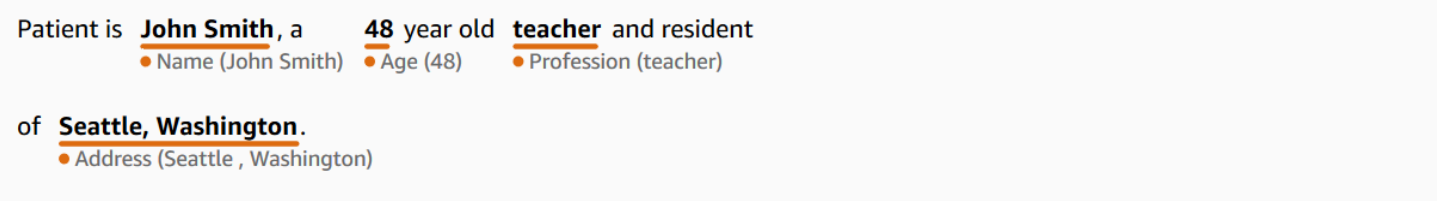
实体	描述	HIPAA 类别
ADDRESS	其中包括任何机构、指定医疗机构或机构内的病房的地址的所有地理细分信息。	2. 地理位置
PROFESSION	包括临床记录中提及的与患者或患者家属有关的任何职业或雇主。	18。任何其他识别特征

示例

文本“病人名叫 John Smith，是一位 48 岁的教师，居住在华盛顿州西雅图。” 返回：

- “John Smith”是 PROTECTED_HEALTH_INFORMATION 类别中的 NAME 类型的实体。
- “48”是 PROTECTED_HEALTH_INFORMATION 类别中的 AGE 类型的实体。
- “教师”是 PROTECTED_HEALTH_INFORMATION 类别中的 PROFESSION 类型（识别特征）的实体。
- “华盛顿州西雅图”是 PROTECTED_HEALTH_INFORMATION 类别中的 ADDRESS 实体。

在 Amazon Comprehend Medical 控制台中，其显示方式如下：



使用 DetectPHI 操作时，响应如下所示。当您使用 “开始作PHIDetection业” 操作时，Amazon Comprehend Medical 会在输出位置创建一个具有这种结构的文件。

```
{
    "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": []
}
```

批量文本分析 APIs

使用 Amazon Comprehend Medical 分析存储在 Amazon S3 桶中的医学文本。一个批次最多分析 10 GB 文档。您可以使用控制台创建和管理批量分析作业，或者使用批处理 APIs 来检测医疗实体，包括受保护的健康信息 (PHI)。 APIs 启动、停止、列出和描述正在进行的批量分析作业。

可以在此处查看批量分析和其他 Amazon Comprehend Medical 操作的定价信息。

重要提示

Amazon Comprehend Medical 的批量分析操作不能代替专业的医学意见、诊断或治疗。为您的使用案例确定正确的置信度阈值，并在需要高准确度的情况下使用高置信度阈值。对于某些使用案例，结果应由经过适当培训的人力审核人员进行审核和验证。Amazon Comprehend Medical 的所有操作只能在患者护理场景中使用，并且要由训练有素的医学专家审查准确性并进行合理的医学判断。

使用执行批量分析 APIs

你可以使用 Amazon Comprehend Medical 控制台或亚马逊 Comprehend Medical Batch 运行批量分析作业。 APIs

先决条件

在使用 Amazon Comprehend Medical API 时，请创建 AWS Identity Access and Management (IAM) 策略并将其附加到 IAM 角色。要了解有关 IAM 角色和信任策略的更多信息，请参阅 IAM 策略和权限。

1. 将数据上传到 S3 桶。
2. 要启动新的分析作业，请使用 StartEntitiesDetection V2Job 操作或 “启动作PHIDetection业” 操作。开始作业时，请告知 Amazon Comprehend Medical 包含输入文件的输入 S3 桶的名称，并指定输出 S3 存储桶以便在批量分析后写入文件。
3. 使用控制台、 DescribeEntitiesDetection V2Job 操作或 Describe Job 操作监控PHIDetection任务的进度。此外， ListEntitiesDetectionV2Jobs 和 List PHIDetection Job 还允许您查看所有本体关联批量分析作业的状态。
4. 如果您需要停止正在进行的作业，请使用 StopEntitiesDetection V2Job 或 Stop PHIDetection Job 停止分析。

5. 要查看分析作业的结果，请参阅您在启动作业时配置的输出 S3 桶。

使用控制台执行批量分析

1. 将数据上传到 S3 桶。
2. 要开始新的分析作业，请选择要执行的分析类型。然后，提供包含输入文件的 S3 桶的名称以及您想要向其发送输出文件的 S3 桶的名称。
3. 在作业运行期间监控其状态。在控制台中，您可以查看所有批量分析操作及其状态，包括分析的开始和结束时间。
4. 要查看分析作业的结果，请查看您在启动作业时配置的输出 S3 桶。

适用于批量操作的 IAM 策略

调用 Amazon Comprehend Medical APIs 批次的 IAM 角色必须具有授予对包含输入和输出文件的 S3 存储桶的访问权限的策略。还必须分配信任关系，让 Amazon Comprehend Medical 服务能够担任该角色。要了解有关 IAM 角色和信任策略的更多信息，请参阅 IAM 角色。

该角色必须具有以下策略。

```
{
    "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"
        }
    ]
}
```

该角色必须拥有以下信任关系。建议您使用 `aws:SourceAccount` 和 `aws:SourceArn` 条件键来防止混淆代理安全问题。要详细了解混淆的代理问题以及如何保护您的 AWS 账户，请参阅 IAM 文档中的混淆代理问题。

```
{
   "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:region:account_id:*"
            }
         }
      }
```

```
    ]
}
```

批量分析输出文件

Amazon Comprehend Medical 为批次中的每个输入文件创建一个输出文件。文件的扩展名为 `.out`。Amazon Comprehend Medical 首先使用 *`AwsAccountId JobType JobId`*-作为名称在输出 S3 存储桶中创建一个目录，然后将该批次的所有输出文件写入该目录。Amazon Comprehend Medical 创建这个新目录的目的是防止一个作业的输出被另一个作业的输出覆盖。

批处理操作产生的输出与同步操作的输出相同。有关 Amazon Comprehend Medical 生成的输出示例，请参阅检测实体（版本 2）。

每个批处理操作都会生成三个清单文件，其中包含有关作业的信息。

- `Manifest` – 总结作业。提供有关作业所用参数、作业的总大小以及已处理的文件数量的信息。
- `success` – 提供有关已成功处理的文件的信息。包括输入和输出文件名以及输入文件的大小。
- `unprocessed` – 列出批处理作业未处理的文件，包括每个文件的错误代码和错误消息。

Amazon Comprehend Medical 会将文件写入您为批处理作业指定的输出目录。摘要清单文件将与标题为 `Manifest_AccountId-Operation-JobId` 的文件夹一起写入输出文件夹。清单文件夹中有一个 `success` 文件夹，其中包含成功清单。还包括一个 `failed` 文件夹，其中包含未处理的文件清单。以下部分显示了清单文件的结构。

批处理清单文件

以下是批处理清单文件的 JSON 结构。

```
{"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"
    }
}
```

成功清单文件

以下是文件的 JSON 结构，其中包含有关已成功处理的文件的信息。

```
{
        "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
        }]
}
```

未处理清单文件

以下是清单文件的 JSON 结构，其中包含有关未处理文件的信息。

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

本体关联

使用 Amazon Comprehend Medical 检测临床文本中的实体，并将这些实体与标准化医学本体中的概念关联起来，包括、ICD-10-CM 和 SNOMED CT RxNorm 知识库。您可以对单个文件进行分析，也可以对大型文档或存储在 Amazon Simple Storage Service (S3) 中的多个文件进行批量分析。

ICD-10-CM 关联

使用Infer ICD1 0CM将可能的疾病作为实体进行检测，并将其与2024年版《国际疾病分类，第10修订版，临床修改》（ICD-10-CM）中的代码关联起来。ICD-10-CM 由美国疾病控制和预防中心 (CDC) 发布。

检测到医学状况时，`InferICD10CM` 会返回匹配的 ICD-10-CM 代码和描述。检测到的状况按置信度降序列出。分数可以表明对与文本中概念相匹配的实体的准确性置信度。家族史、体征、症状和否定等相关信息被视为特征。解剖学名称和猛烈程度等其他信息被列为属性。

Infer ICD1 0CM 非常适合以下场景：

- 协助对患者病历进行专业医学编码
- 临床研究和试验
- 与医疗软件系统集成
- 早期发现和诊断
- 人口健康管理

ICD-10-CM 类别

Infer ICD1 0CM 会检测该类别中的`MEDICAL_CONDITION`实体。它还会检测其他相关信息，并将其作为属性或特征来关联。

ICD-10-CM 类型

Infer ICD1 0CM 可检测到以下类型的`DX_NAME`实体：和。`TIME_EXPRESSION`

ICD-10-CM 特征

Infer ICD1 0CM 将以下上下文信息检测为特征：

- DIAGNOSIS：通过对症状的评估来确定某种医学状况。
- HYPOTHETICAL：表明医学状况是一种假设。
- LOW_CONFIDENCE：表明医学状况具有高度不确定性。这与提供的置信度分数没有直接关系。
- NEGATION：表明不存在医学状况。
- PERTAINS_TO_FAMILY：表明医学状况与患者家属有关，与患者无关。
- SIGN：医生报告的医学状况。
- SYMPTOM：患者报告的医学状况。

ICD-10-CM 属性

Infer ICD1 0CM 将以下上下文信息检测为属性：

- DIRECTION：方位词。例如左、右、内侧、外侧、上、下、后、前、远端、近端、对侧、双侧、同侧、背侧或腹侧。
- SYSTEM_ORGAN_SITE：解剖学位置。
- ACUITY:：确定疾病的猛烈程度，例如慢性、急性、突发性、持续性或渐进性。其仅适用于 MEDICAL_CONDITION 类型。
- QUALITY：医学状况的任何描述词语，例如阶段或等级。

时间类别

TIME_EXPRESSION 类别可以检测与时间相关的实体。其中包括日期和时间等实体，例如 “三天前”、“今天”、“当前”、“入院当天”、“上个月”或“16 天”。此类别中的结果仅在与实体关联时才会返回。例如，“患者昨天被诊断出患有流感”会将 Yesterday 作为与 DX_NAME 实体“流感”重叠的 TIME_EXPRESSION 实体返回。但是，在“患者昨天遛过狗”中，“昨天”不会被认定为一个实体。

类型

识别出的 TIME_EXPRESSION 类型为 TIME_TO_DX_NAME：医学状况的发生日期。此类型的属性为 DX_NAME。

关系类型

RELATIONSHIP_TYPE 是指实体与属性之间的关系。识别出的 RELATIONSHIP_TYPE 是 OVERLAP – 与检测到的实体一致的 TIME_EXPRESSION。

输入和响应示例

> **Note**
>
> 有关具体的 API 输入和响应语法，请参阅《亚马逊 Comprehend Medical API 参考》中的 InferICD1 0CM。

以下示例输入文本显示了 InferICD10CM 操作的工作原理。要查看所有输入文本，请滚动复制按钮。

```
"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 操作以 JSON 格式返回以下输出（为简洁起见，采用缩写）。

```
{
    "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": "3.0.0.20231001"
}
```

InferICD10CM 还能识别文本中的否定实体。例如，如果患者没有出现某种症状，则症状和否定均会被识别为特征，并列出置信度分数。对于前一个示例的输入，症状 Nausea 将列于 NEGATION 下方，因为患者没有出现恶心。

```
{
    "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 链接

使用该InferRxNorm操作将患者记录中列为实体的药物进行识别。该行动还将这些实体与国家医学图书馆 RxNorm 数据库中的概念标识符（rxCUI）联系起来。每个 rxCUI 的来源都是 2022-11-0 RxNorm 7 和发行版。 RxTerms 每个 RxCUI 都是独一无二的，具有不同的强度和剂量形式。Amazon Comprehend Medical 按置信CUIs 度分数降序列出了其检测到的每种药物的最大可能匹配的处方。非结构化文本无法使用 RxCUI 代码进行下游分析。强度、频率、剂量、剂量形式和给药途径等相关信息以 JSON 格式列为属性。

您可以将 InferRxNorm 用于以下场景：

- 筛查患者服用的药物。
- 防止新开的药物与患者目前正在服用的药物之间出现潜在的不良反应。
- 使用 RxCUI 根据用药史筛查是否纳入临床试验。
- 检查药物的剂量和频率是否适当。
- 筛查药物的用途、适应症和副作用。
- 管理人口健康。

重要提示

Amazon Comprehend Medical的InferRxNorm运营并不能替代专业的医疗建议、诊断或治疗。为您的使用案例确定正确的置信度阈值，并在需要高准确度的情况下使用高置信度阈值。只有经过训练有素的医学专家审查结果的准确性和进行合理的医学判断后，才能在患者护理场景中使用 Amazon Comprehend Medical。

RxNorm 类别

InferRxNorm检测`MEDICATION`类别中的实体。它还会检测作为属性或特征来关联的其他相关信息。

RxNorm 类型

`Medication` 类别中的实体类型为：

- `BRAND_NAME`：药物或治疗剂的受版权保护的品牌名称。
- `GENERIC_NAME`：药物或治疗剂的非品牌名称、成分名称或配方混合物。

RxNorm 属性

- `DOSAGE`：医嘱用药量。
- `DURATION`：用药持续时间。
- `FORM`：药物的形式。
- `FREQUENCY`：用药频率。
- `RATE`: 药物的给药率（主要用于药物输液或 IVs）。
- `ROUTE_OR_MODE`：药物的用药方法。

- STRENGTH：药物强度。

RxNorm 特征

- NEGATION：表明患者没有用药。
- PAST_HISTORY：表明检测到的药物来自患者的过去（现在之前）。

输入和响应示例

> **Note**
>
> 有关具体的 API 输入和响应语法，请参阅InferRxNorm《亚马逊 Comprehend Medical API 参考》。

以下示例输入文本显示了 InferRxNorm 操作的工作原理。要查看所有输入文本，请滚动复制按钮。

```
"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 操作以 JSON 格式返回以下输出：

```
{
    "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": "3.0.0.20221107"
}
```

使用以下输入文本，InferRxNorm 操作也可以识别否定特征。

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

InferRxNorm 操作以 JSON 格式返回以下输出：

```
{
    "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": "3.0.0.20221107"
}
```

SNOMED CT 关联

使用 InferSNOMEDCT 操作可以检测医学实体，并将其与 2022 年 3 月版本的《医学系统术语表 (SNOMED CT)》中的概念关联起来。SNOMED CT 提供了全面的医学概念词汇，包括医学状况和解剖学、医学检查、治疗和手术。要了解有关 SNOMED CT 的更多信息，请访问 SNOMED CT。

对于每个检测到的医疗实体，Amazon Comprehend Medical 会列出前五名 SNOMED CT IDs 概念和与医学概念相关的描述，以及表明模型对其预测可信度的置信度分数。SNOMED CT 概念 IDs 与置信度分数一起按置信度降序列出。然后，当您将 SNOMED CT 概念与 SNOMED CT 多元层次结构一起使用时， IDs 可以使用 SNOMED CT 概念来构建患者临床数据，用于医学编码、报告或临床分析。

InferSNOMEDCT 可供美国客户使用。有关其他国家/地区的 SNOMED CT 以及 SNOMED CT 许可的信息，请参阅 SNOMED CT。

InferSNOMEDCT 非常适合以下场景：

- 协助对患者病历进行专业医学编码
- 临床研究和试验
- 人口健康管理

InferSNOMEDCT 可以检测以下类别的实体：它还会检测其他上下文信息，并将其作为属性或特征来关联。

- `MEDICAL_CONDITION`：医学状况的体征、症状和诊断。
- `ANATOMY`：身体部位或身体系统以及这些部位或系统的位置。
- `TEST_TREATMENT_PROCEDURE`：用于确定医学状况的程序。

解剖学类别

`ANATOMY` 类别可以检测对身体部位或身体系统以及这些部位或系统的位置的表述。

Attributes

可以检测 `ANATOMY` 类别的以下属性：

- `DIRECTION`：方位词。例如左、右、内侧、外侧、上、下、后、前、远端、近端、对侧、双侧、同侧、背侧或腹侧。
- `SYSTEM_ORGAN_SITE`：身体系统、解剖学位置或区域以及身体部位。

医学状况类别

MEDICAL_CONDITION 类别可以检测医学状况的体征、症状和诊断。

类型

对于 MEDICAL_CONDITION 类别，可以检测以下类型：

- DX_NAME:：通过对症状的评估来确定某种医学状况。

Attributes

可以检测 MEDICAL_CONDITION 类别的以下属性：

- ACUITY:：确定疾病的猛烈程度，例如慢性、急性、突发性、持续性或渐进性。
- QUALITY:：医学状况的任何描述词语，例如阶段或等级。
- DIRECTION：方位词。例如左、右、内侧、外侧、上、下、后、前、远端、近端、对侧、双侧、同侧、背侧或腹侧。
- SYSTEM_ORGAN_SITE：身体系统、解剖学位置或区域以及身体部位。

特征

可以检测 MEDICAL_CONDITION 类别的以下特征：

- DIAGNOSIS：被确定为症状的原因或结果的医学状况。症状可以通过体检结果、实验室或放射学报告或其他方式发现。
- HYPOTHETICAL：表明医学状况是一种假设。
- LOW_CONFIDENCE：表明医学状况具有高度不确定性。这与提供的置信度分数没有直接关系。
- NEGATION：表明不存在医学状况。
- PERTAINS_TO_FAMILY：表明医学状况与患者家属有关，与患者无关。
- SIGN：医生报告的医学状况。
- SYMPTON：患者报告的医学状况。

检查、治疗和手术类别

TEST_TREATMENT_PROCEDURE 类别可以检测用于确定医学状况的程序。

类型

对于 TEST_TREATMENT_PROCEDURE 类别，可以检测以下类型：

- PROCEDURE_NAME:：对患者进行的干预措施，用于治疗疾病或为患者提供护理。
- TEST_NAME:：对患者执行的诊断、测量、筛查或评级程序，最后可能产生结果数值。其中包括用于确定诊断、排除或确定病情或对患者进行评级或打分的任何程序、流程、评估或评分。
- TREATMENT_NAME:：为对抗疾病或失调而采取的干预措施。其中包括药物，例如抗病毒药物和疫苗接种。

Attributes

对于 TEST_TREATMENT_PROCEDURE 类别，可以检测以下属性：

- TEST_NAME:：进行过的诊断检查。
- TEST_VALUE:：诊断检查的数字结果。
- TEST_UNIT:：与 TEST_VALUE: 结果相关的单位。
- PROCEDURE_NAME:：进行过的手术或医疗程序的名称。
- TREATMENT_NAME:：对患者实施的治疗的名称。

特征

- FUTURE：表明检查、治疗或手术是将在临床记录中的主题事件之后发生的行动或事件。
- HYPOTHETICAL：表明检查、治疗或手术是一种假设
- NEGATION：表明结果为阴性或行动未执行。
- PAST_HISTORY：表明检查、治疗或手术来自患者的过去（现在之前）。

SNOMED CT 详细信息

JSON 响应中包含 SNOMED CT 详细信息，其中包括以下信息：

- EDITION:：仅支持美国版。
- VERSIONDATE: ：使用的 SNOMED CT 版本的日期戳。
- LANGUAGE:：支持对英语 (US-EN) 进行分析。

输入和响应示例

> **Note**
>
> 有关具体的 API 输入和响应语法，请参阅《Amazon Comprehend Medical API 参考》中的 InferSNOMEDCT。

以下示例输入文本显示了 InferSNOMEDCT 操作的工作原理。要查看所有输入文本，请滚动复制按钮。

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

InferSNOMEDCT 操作以 JSON 格式返回以下输出。

```
{
    "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": "3.0.0.20220301"
}
```

本体关联批量分析

使用 Amazon Comprehend Medical 检测存储在 Amazon Simple Storage Service (Amazon S3) 桶中的临床文本实体，并将这些实体关联到标准化本体。您可以使用本体关联批量分析来分析一组文档或最多包含 20000 个字符的单个文档。通过使用控制台或本体关联批处理 API，您可以执行操作，以启动、停止、列出和描述正在进行的批量分析作业。

有关批量分析和其他 Amazon Comprehend Medical 操作的定价信息，请参阅 Amazon Comprehend Medical Pricing。

执行批量分析

您可以使用 Amazon Comprehend Medical 控制台或 Amazon Comprehend Medical 批处理 API 操作来运行批量分析作业。

使用 API 操作执行批量分析

先决条件

在使用 Amazon Comprehend Medical API 时，请创建 AWS Identity Access and Management (IAM) 策略并将其附加到 IAM 角色。要了解有关 IAM 角色和信任策略的更多信息，请参阅 IAM 策略和权限。

1. 将数据上传到 S3 桶。
2. 要启动新的分析作业，请使用 Start ICD1 0 CMInference Job、Start SNOMEDCTInference Job 或StartRxNormInferenceJob操作。提供包含输入文件的 Amazon S3 桶的名称以及您将要向其发送输出文件的 Amazon S3 桶的名称。

3. 使用 Desc ribe ICD1 0 Job、Describe CMInference Jo b 或DescribeRxNormInferenceJob操作来监控SNOMEDCTInference任务的进度。此外，您可以使用列出 ICD1 0 CMInference 作业、列出SNOMEDCTInference作业以及ListRxNormInferenceJobs查看所有本体关联批量分析作业的状态。
4. 如果您需要停止正在进行的作业，请使用 Stop ICD1 0 CMInference Job、Stop SNOMEDCTInference J ob 或StopRxNormInferenceJob停止分析。
5. 要查看分析作业的结果，请参阅您在启动作业时配置的输出 S3 桶。

使用控制台执行批量分析

1. 将数据上传到 S3 桶。
2. 要开始新的分析作业，请选择要执行的分析类型。然后，提供包含输入文件的 S3 桶的名称以及您将要向其发送输出文件的 S3 桶的名称。
3. 在作业运行期间监控其状态。在控制台中，您可以查看所有批量分析操作及其状态，包括分析的开始和结束时间。
4. 要查看分析作业的结果，请查看您在启动作业时配置的输出 S3 桶。

适用于批量操作的 IAM 策略

调用 Amazon Comprehend Medical 批处理 API 操作的 IAM 角色必须有一个策略，允许访问包含输入和输出文件的 S3 桶。此外，还必须为 IAM 角色分配信任关系，这样 Amazon Comprehend Medical 服务才能担任该角色。要了解有关 IAM 角色和信任策略的更多信息，请参阅 IAM 角色。

该角色必须具有以下策略：

```
{
    "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"
        }
    ]
}
```

该角色必须拥有以下信任关系。建议您使用 `aws:SourceAccount` 和 `aws:SourceArn` 条件键来防止混淆代理安全问题。要详细了解混淆的代理问题以及如何保护您的 AWS 账户，请参阅 IAM 文档中的混淆代理问题。

```
{
   "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:region:account_id:*"
            }
        }
    }
  ]
}
```

批量分析输出文件

Amazon Comprehend Medical 为批次中的每个输入文件创建一个输出文件。文件的扩展名为 `.out`。Amazon Comprehend Medical 首先使用 *`AwsAccountId JobType JobId`*-作为名称在输出 S3 存储桶中创建一个目录，然后将该批次的所有输出文件写入该目录。Amazon Comprehend Medical 创建此新目录，这样一个作业的输出就不会覆盖另一个作业的输出。

批处理操作产生的输出与同步操作相同。

每个批处理操作都会生成以下三个清单文件，其中包含有关作业的信息：

- `Manifest` – 总结作业。提供有关作业所用参数、作业的总大小以及已处理的文件数量的信息。
- `Success` – 提供有关已成功处理的文件的信息。包括输入和输出文件名以及输入文件的大小。
- `Unprocessed` – 列出批处理作业未处理的文件，每个文件都有错误代码和错误消息。

Amazon Comprehend Medical 会将文件写入您为批处理作业指定的输出目录。摘要清单文件将与标题为 `Manifest_AccountId-Operation-JobId` 的文件夹一起写入输出文件夹。清单文件夹中有 `success` 文件夹和 `failed` 文件夹，前者包含成功清单，后者包含未处理的文件清单。以下部分显示了清单文件的结构。

批处理清单文件

以下是批处理清单文件的 JSON 结构。

```
{"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"
    }
}
```

成功清单文件

以下是文件的 JSON 结构，其中包含有关已成功处理的文件的信息。

```
{
    "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
     }]
}
```

未处理清单文件

以下是清单文件的 JSON 结构，其中包含有关未处理文件的信息。

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

```
  },
  { ...}
  ]
}
```

Amazon Comprehend Medical 中的安全性

云安全 AWS 是重中之重。作为 AWS 客户，您可以受益于专为满足大多数安全敏感型组织的要求而构建的数据中心和网络架构。

安全是双方 AWS 的共同责任。责任共担模式将其描述为云的安全性和云中的安全性：

- 云安全 — AWS 负责保护在 AWS 云中运行 AWS 服务的基础架构。 AWS 还为您提供可以安全使用的服务。作为AWS 合规计划合规计划合规计划合的一部分，第三方审计师定期测试和验证我们安全的有效性。要了解适用于 Amazon Comprehend Medical 的合规性计划，请参阅按合规性计划提供的范围内 AWS 服务。
- 云端安全-您的责任由您使用的 AWS 服务决定。您还需要对其他因素负责，包括您的数据的敏感性、您公司的要求以及适用的法律法规。

该文档帮助您了解如何在使用 Comprehend Medical 时应用责任共担模式。以下主题说明如何配置 Comprehend Medical 以实现您的安全性和合规性目标。您还将了解如何使用其他 AWS 服务来帮助您监控和保护 Comprehend Medical 资源。

主题

- Amazon Comprehend Medical 中的数据保护
- Amazon Comprehend Medical 中的身份和访问管理
- 使用记录亚马逊 Comprehend Medical API 调用 AWS CloudTrail
- Amazon Comprehend Medical 的合规性验证
- Amazon Comprehend Medical 中的弹性
- Amazon Comprehend Medical 中的基础设施安全性

Amazon Comprehend Medical 中的数据保护

分担责任模型 AWS 分担责任模型适用于 Amazon Comprehend Medical 中的数据保护。如本模型所述 AWS ，负责保护运行所有内容的全球基础架构 AWS Cloud。您负责维护对托管在此基础结构上的内容的控制。您还负责您所使用的 AWS 服务 的安全配置和管理任务。有关数据隐私的更多信息，请参阅数据隐私常见问题。有关欧洲数据保护的信息，请参阅 AWS Security Blog 上的 AWS Shared Responsibility Model and GDPR 博客文章。

出于数据保护目的，我们建议您保护 AWS 账户 凭证并使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 设置个人用户。这样，每个用户只获得履行其工作职责所需的权限。还建议您通过以下方式保护数据：

- 对每个账户使用多重身份验证（MFA）。
- 使用 SSL/TLS 与资源通信。 AWS 我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 使用设置 API 和用户活动日志 AWS CloudTrail。有关使用 CloudTrail 跟踪捕获 AWS 活动的信息，请参阅AWS CloudTrail 用户指南中的使用跟 CloudTrail 踪。
- 使用 AWS 加密解决方案以及其中的所有默认安全控件 AWS 服务。
- 使用高级托管安全服务（例如 Amazon Macie），它有助于发现和保护存储在 Amazon S3 中的敏感数据。
- 如果您在 AWS 通过命令行界面或 API 进行访问时需要经过 FIPS 140-3 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅《美国联邦信息处理标准（FIPS）第 140-3 版》。

强烈建议您切勿将机密信息或敏感信息（如您客户的电子邮件地址）放入标签或自由格式文本字段（如名称字段）。这包括你使用控制台、API 或与 Comprehend Medical AWS 服务 或其他机构合作时。 AWS CLI AWS SDKs在用于名称的标签或自由格式文本字段中输入的任何数据都可能会用于计费或诊断日志。如果您向外部服务器提供网址，强烈建议您不要在网址中包含凭证信息来验证对该服务器的请求。

Amazon Comprehend Medical 中的身份和访问管理

访问 Comprehend Medical 时需要使用凭证，以便 AWS 用来验证您的请求。这些凭证必须具有访问 Comprehend Medical操作的权限。AWS Identity and Access Management (IAM) 可以控制能够访问您的资源的用户，从而保护这些资源。以下部分详细信息说明了如何在 Comprehend Medical 中使用 IAM。

- 身份验证
- 访问控制

身份验证

您必须向用户授予与 Amazon Comprehend Medical 互动的权限。对于需要完全访问权限的用户，请使用 `ComprehendMedicalFullAccess`。

要提供访问权限，请为您的用户、组或角色添加权限：

- 中的用户和群组 AWS IAM Identity Center：

  创建权限集合。按照《AWS IAM Identity Center 用户指南》中创建权限集的说明进行操作。
- 通过身份提供商在 IAM 中托管的用户：

  创建适用于身份联合验证的角色。按照《IAM 用户指南》中针对第三方身份提供商创建角色（联合身份验证）的说明进行操作。
- IAM 用户：
  - 创建您的用户可以担任的角色。按照《IAM 用户指南》中为 IAM 用户创建角色的说明进行操作。
  - （不推荐使用）将策略直接附加到用户或将用户添加到用户组。按照《IAM 用户指南》中向用户添加权限（控制台）中的说明进行操作。

要使用 Amazon Comprehend Medical 的异步操作，您还需要一个服务角色。

服务角色是由一项服务担任、代表您执行操作的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

要详细了解如何将 Amazon Comprehend Medical 指定为主体服务，请参阅批量操作所需的基于角色的权限。

访问控制

您必须通过有效的凭证对您的请求进行身份验证。这些凭证必须具有调用 Amazon Comprehend Medical 操作的权限。

以下部分介绍了如何管理 Amazon Comprehend Medical 的权限。我们建议您先阅读概述。

- 管理 Amazon Comprehend Medical 资源的访问权限的概述
- 将基于身份的策略（IAM 策略）用于 Amazon Comprehend Medical

主题

- 管理 Amazon Comprehend Medical 资源的访问权限的概述
- 将基于身份的策略（IAM 策略）用于 Amazon Comprehend Medical
- Amazon Comprehend Medical API 权限：操作、资源和条件参考
- AWS 亚马逊 Comprehend Medical 的托管政策

管理 Amazon Comprehend Medical 资源的访问权限的概述

权限策略可以管理对某项操作的访问权限。账户管理员可以将权限策略附加到 IAM 身份，从而管理对操作的访问权限。IAM 身份包括用户、组和角色。

> **Note**
>
> 账户管理员（或管理员用户）是具有管理员权限的用户。有关更多信息，请参阅《IAM 用户指南》中的 IAM 最佳实操。

在授予权限时，您可以决定谁和哪项操作可以获得权限。

主题

- 管理对操作的访问权限
- 指定策略元素：操作、效果和主体
- 在策略中指定条件

管理对操作的访问权限

权限策略规定谁可以访问哪些内容。以下部分介绍了适用于权限策略的选项。

> **Note**
>
> 本节介绍 Amazon Comprehend Medical 中的 IAM。这里不提供有关 IAM 服务的详细信息。有关 IAM 的更多信息，请参阅《IAM 用户指南》中的什么是 IAM？。有关 IAM 策略语法和说明的信息，请参阅《IAM 用户指南》中的 AWS IAM 策略参考。

附加到 IAM 身份的策略是基于身份的策略。附加到资源的策略是基于资源的策略。Amazon Comprehend Medical 只支持基于身份的策略。

基于身份的策略（IAM 策略）

您可以向 IAM 身份附加策略。以下是两个示例。

- 将权限策略附加到账户中的用户或组。要允许某个用户或一组用户调用 Amazon Comprehend Medical 操作，请将权限策略附加到用户。请将策略附加到包含该用户的组。

- 将权限策略附加到角色以便授予跨账户权限。要授予跨账户权限，请将基于身份的权限策略附加到 IAM 角色。例如，账户 A 中的管理员可以创建一个角色，以便向另一账户授予跨账户权限。在本示例中，我们将其称为账户 B，它也可以是一项 AWS 服务。
  1. 账户 A 管理员创建一个 IAM 角色，并向该角色附加一项策略，授权其访问账户 A 中的资源的权限。
  2. 账户 A 管理员将信任策略附加到角色。该策略将账户 B 标识为可担任该角色的主体。
  3. 之后，账户 B 管理员可以委派权限，指派账户 B 中的任何用户担任该角色。这样，账户 B 中的用户就可以在账户 A 中创建或访问资源。如果您需要向某项 AWS 服务授予权限以便担任该角色，则信任策略中的主体也可以是 AWS 服务主体。

  有关使用 IAM 委托权限的更多信息，请参阅 IAM 用户指南中的访问权限管理。

有关将基于身份的策略与 Amazon Comprehend Medical 结合使用的更多信息，请参阅将基于身份的策略（IAM 策略）用于 Amazon Comprehend Medical。有关用户、组、角色和权限的更多信息，请参阅《IAM 用户指南》https://docs.aws.amazon.com/IAM/latest/UserGuide/id.html中的身份（用户、组和角色）。

基于资源的策略

其他服务（例如） AWS Lambda支持基于资源的权限策略。例如，您可以将策略附加到 S3 存储桶以管理对该存储桶的访问权限。Amazon Comprehend Medical 不支持基于资源的策略。

指定策略元素：操作、效果和主体

Amazon Comprehend Medical 定义了一组 API 操作。为授予对这些 API 操作的权限，Amazon Comprehend Medical 定义了一组您可以在策略中指定的操作。

以下四个项目是最基本的策略元素。

- 资源 – 在策略中，您可以使用 Amazon Resource Name (ARN) 来标识策略应用到的资源。对于 Amazon Comprehend Medical，资源永远是 `"*"`。
- 操作 – 使用操作关键字来标识要允许或拒绝的操作。例如，根据指定的效果，`comprehendmedical:DetectEntities` 可以允许或拒绝执行 Amazon Comprehend Medical `DetectEntities` 操作的用户权限。
- 效果 – 指定当用户请求特定操作时出现的效果（允许或拒绝）。如果没有显式授予（允许）对资源的访问权限，则隐式拒绝访问。您也可显式拒绝对资源的访问。您可以执行此操作，以确保用户无法访问资源，即使有其他策略授予了访问权限也是如此。

- 主体 – 在基于身份的策略中，附加了策略的用户是隐式主体。

有关 IAM 策略语法和介绍的更多信息，请参阅 IAM 用户指南中的 AWS IAM 策略参考。

有关显示所有 Amazon Comprehend Medical API 操作的表，请参阅 Amazon Comprehend Medical API 权限：操作、资源和条件参考。

在策略中指定条件

授予权限时，您可以使用 IAM 策略语言指定策略生效的条件。例如，您可能希望策略仅在特定日期后应用。有关使用策略语言指定条件的更多信息，请参阅《IAM 用户指南》中的条件。

AWS 为支持通过 IAM 进行访问控制的所有 AWS 服务提供了一组预定义条件键。例如，您可以使用 `aws:userid` 条件键要求用户在请求某项操作时提供特定的 AWS ID。有关更多信息和 AWS 键的完整列表，请参阅 IAM 用户指南中的可用条件键。

Amazon Comprehend Medical 不提供任何额外的条件键。

将基于身份的策略（IAM 策略）用于 Amazon Comprehend Medical

本主题介绍基于身份的策略示例。示例展示了账户管理员如何向 IAM 身份附加权限策略。这让用户、组和角色能够执行 Amazon Comprehend Medical 操作。

> **Important**
>
> 要了解权限，我们建议您查看管理 Amazon Comprehend Medical 资源的访问权限的概述。

以下示例策略是使用 Amazon Comprehend Medical 文档分析操作所需的策略。

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

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

                "comprehendmedical:StartPHIDtectionJob",
                "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": "*"
      }
   ]
}
```

策略中的一个语句授予使用 `DetectEntities` 和 `DetectPHI` 操作的权限。

该策略不指定 `Principal` 元素，因为在基于身份的策略中，未指定获取权限的委托人。将策略附加到用户时，该用户是隐式主体。向 IAM 角色附加策略后，该角色的信任策略中标识的主体将获取权限。

要查看所有 Amazon Comprehend Medical API 操作及其适用的资源，请参阅 Amazon Comprehend Medical API 权限：操作、资源和条件参考。

使用 Amazon Comprehend Medical 控制台所需的权限

权限参考表列出了 Amazon Comprehend Medical API 操作，并显示了每个操作所需的权限。有关 Amazon Comprehend Medical API 权限的更多信息，请参阅 Amazon Comprehend Medical API 权限：操作、资源和条件参考。

要使用 Amazon Comprehend Medical 控制台，请为以下策略中所示的操作授予权限。

```
{
  "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 控制台出于以下原因需要以下权限：

- `iam` 权限，用于列出您的账户的可用 IAM 角色。
- `s3` 权限，用于访问包含数据的 Amazon S3 桶和对象。

使用控制台创建异步批处理作业时，您还可以为作业创建 IAM 角色。要使用控制台创建 IAM 角色，用户必须获得以下附加权限才能创建 IAM 角色和策略，并将策略附加到角色。

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

Amazon Comprehend Medical 控制台需要这些权限才能创建角色和策略并附加角色和策略。`iam:PassRole` 操作让控制台能够将角色传递给 Amazon Comprehend Medical。

适用于 Amazon Comprehend Medical 的 AWS 托管（预定义）策略

AWS 可以提供由 AWS 创建和管理的独立多种 IAM 策略，用来处理各种常见使用场景。这些 AWS 托管策略可以针对常见使用场景授予必要的权限，让您无需研究需要哪些权限。有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管策略。

可以附加到您账户中的用户的以下 AWS 托管式策略专门适用于 Amazon Comprehend Medical。

- ComprehendMedicalFullAccess— 授予对亚马逊 Comprehend Medical 资源的完全访问权限。包括列出和获取 IAM 角色的权限。

您必须向使用 Amazon Comprehend Medical 的任何用户应用以下附加策略：

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

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

您可以通过登录到 IAM 控制台并在该控制台中搜索特定策略来查看托管权限策略。

这些策略在您使用 AWS SDKs 或 AWS CLI 时起作用。

您还可以创建您自己的 IAM 策略，以便授予对 Amazon Comprehend Medical 操作和资源的权限。您可以将这些自定义策略附加到需要它们的 IAM 用户或组。

批量操作所需的基于角色的权限

要使用 Amazon Comprehend Medical 异步操作，请向 Amazon Comprehend Medical 授予对包含文档集合的 Amazon S3 桶的访问权限。为此，请在账户中创建一个数据访问角色，并使其信任 Amazon Comprehend Medical 服务主体。有关创建角色的更多信息，请参阅 AWS Identity and Access Management 用户指南中的创建向 AWS 服务委派权限的角色。

以下是该角色的信任策略。

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

创建角色后，请为该角色创建访问策略。该策略应该授予对包含您的输入数据的 Amazon S3 桶的 Amazon S3 `GetObject` 和 `ListBucket` 权限。它还会授予对您的 Amazon S3 输出数据桶的 Amazon S3 `PutObject` 权限。

以下示例访问策略包含这些权限。

```
{
    "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"
        }
    ]
}
```

客户管理型策略示例

本节的用户策略示例介绍如何授予对各种 Amazon Comprehend Medical 操作的权限。这些策略在您使用 AWS SDKs 或 AWS CLI 时起作用。使用控制台时，必须向所有亚马逊 Comprehend Medical 授予权限。 APIs这在 使用 Amazon Comprehend Medical 控制台所需的权限 中进行了讨论。

> **Note**
>
> 所有示例都使用 us-east-2 区域并包含虚构账户。 IDs

示例

示例 1：允许所有 Amazon Comprehend Medical 操作

注册后 AWS，您可以创建管理员来管理您的账户，包括创建用户和管理他们的权限。

您可以创建对所有 Amazon Comprehend 操作都具有权限的用户。您可以将该用户视为特定于服务的管理员，为使用 Amazon Comprehend 而设置。您可以将以下权限策略附加到该用户。

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

示例 2：仅允许 DetectEntities操作

以下权限策略授予用户检测 Amazon Comprehend Medical 中的实体的权限，但不允许用户检测 PHI 操作。

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Sid": "AllowDetectEntityActions",
```

```
        "Effect": "Allow",
        "Action": [
                  "comprehendedical:DetectEntities"
              ],
             "Resource": "*"
             ]
          }
      ]
}
```

Amazon Comprehend Medical API 权限：操作、资源和条件参考

在设置 访问控制 和编写可以附加到用户的权限策略时，您可以参考下表。此列表包含每个 Amazon Comprehend Medical API 操作、您可以授予执行权限的对应操作和您可以授予权限的 AWS 资源。您可以在策略的 `Action` 字段中指定这些操作，并在策略的 `Resource` 字段中指定资源值。

要表达条件，您可以在 Amazon Comprehend Medical 策略中使用 AWS 条件键。有关键的完整列表，请参阅 IAM 用户指南中的可用键。

> **Note**
>
> 要指定操作，请在 API 操作名称之前使用 `comprehendmedical:` 前缀，例如 `comprehendmedical:DetectEntities`。

AWS 亚马逊 Comprehend Medical 的托管政策

AWS 托管策略是由创建和管理的独立策略 AWS。 AWS 托管策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住， AWS 托管策略可能不会为您的特定用例授予最低权限权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于您的使用场景的客户管理型策略来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新 AWS 托管策略中定义的权限，则更新会影响该策略所关联的所有委托人身份（用户、组和角色）。 AWS 最有可能在启动新的 API 或现有服务可以使用新 AWS 服务 的 API 操作时更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管策略。

主题

- AWS 托管策略： ComprehendMedicalFullAccess
- Comprehend Medical 对托管政策的更新 AWS

AWS 托管策略： ComprehendMedicalFullAccess

您可以将 `ComprehendMedicalFullAccess` 策略附加到 IAM 身份。

该政策向所有 Comprehend Medical 行动授予管理许可。

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

Comprehend Medical 对托管政策的更新 AWS

查看自 Amazon Comprehend Medical 开始跟踪这些变更以来该服务对 AWS 托管政策的更新的详细信息。有关此页面更改的自动提示，请订阅 文档历史记录页面上的 RSS 信息源。

更改	描述	日期
Comprehend Medical 开始追踪变更	Comprehend Medical开始跟踪其托管保单的变更。 AWS	2018 年 11 月 27 日

使用记录亚马逊 Comprehend Medical API 调用 AWS CloudTrail

亚马逊 Comprehend Medical 已与集成。 AWS CloudTrail CloudTrail 是一项服务，用于记录用户、角色或服务在 Amazon Comprehen AWS d Medical 内部采取的操作。 CloudTrail 将 Amazon Comprehend Medical 的所有 API 调用捕获为事件。记录的调用包括来自 Amazon Comprehend Medical 控制台的调用和对 Amazon Comprehend Medical API 操作的代码调用。如果您创建跟踪，则可以允许将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括亚马逊 Comprehend Medical 的事件。如果您未配置跟踪，您仍然可以在 CloudTrail 控制台的 “事件历史记录” 中查看最新的事件。使用收集的信息 CloudTrail，您可以确定以下几点，例如：

- 向 Amazon Comprehend Medical 发出的请求
- 发出请求的 IP 地址
- 谁发出了请求
- 发出请求的时间
- 其他详细信息

要了解更多信息 CloudTrail，请参阅《AWS CloudTrail 用户指南》。

亚马逊 Comprehend Medical 信息在 CloudTrail

CloudTrail 在您创建 AWS 账户时已在您的账户上启用。当 Amazon Comprehend Medical 中发生活动时，该活动会 AWS 与其他服务事件一起记录在 CloudTrail 事件历史记录中。您可以在自己的 AWS 账户中查看、搜索和下载最近发生的事件。有关更多信息，请参阅使用事件历史记录查看 CloudTrail 事件。

要持续记录您的 AWS 账户中的事件，包括亚马逊 Comprehend Medical 的事件，请创建跟踪。跟踪允许 CloudTrail将日志文件传输到 Amazon S3 存储桶。默认情况下，在控制台中创建跟踪记录时，此跟踪记录应用于所有 AWS 区域。跟踪记录 AWS 分区中所有区域的事件，并将日志文件传送到您指定的 Amazon S3 存储桶。此外，您可以配置其他 AWS 服务，以进一步分析和处理 CloudTrail 日志中收集的事件数据。有关更多信息，请参阅下列内容：

- 创建跟踪概述
- CloudTrail 支持的服务和集成
- 配置 Amazon SNS 通知 CloudTrail
- 接收来自多个区域的 CloudTrail 日志文件和接收来自多个账户的 CloudTrail 日志文件

所有亚马逊 Comprehend Medical 操作均 CloudTrail 由亚马逊 Comprehend Medical API 参考记录并记录在案。例如，调用`DetectPHI`和`ListEntitiesDetectionV2Jobs`操作会在 CloudTrail 日志文件中生成条目。`DetectEntitiesV2`

每个事件或日志条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用根证书还是 AWS Identity and Access Management (IAM) 用户凭证发出。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出。

有关更多信息，请参阅 CloudTrail userIdentity 元素。

了解 Amazon Comprehend Medical 日志文件条目

跟踪是一种配置，允许将事件作为日志文件传输到您指定的 Amazon S3 存储桶。 CloudTrail 日志文件包含一个或多个日志条目。事件表示来自任何源的单个请求。该事件包含有关所请求操作的信息，例如日期和时间或请求参数。 CloudTrail 日志文件不是公共 API 调用的有序堆栈跟踪，因此它们不会按任何特定的顺序出现。

以下示例显示了演示该`DetectEntitiesV2`操作的 CloudTrail 日志条目。

```
        {
    "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 的合规性验证

作为多项合规计划的一部分，第三方审计师评估亚马逊Comprehend Medical的安全与合规性。 AWS 其中包括 PCI、FedRAMP、HIPAA 及其他计划。您可以使用下载第三方审计报告 AWS Artifact。有关更多信息，请参阅在 AWS Artifact 中下载报告。

您在使用Comprehend Medical时的合规责任取决于您的数据的敏感性、贵公司的合规目标以及适用的法律和法规。 AWS 提供了以下资源来帮助实现合规性：

- 《安全性与合规性快速入门指南》– 这些部署指南讨论了架构注意事项，并提供了在 AWS上部署基于安全性和合规性的基准环境的步骤。
- HIPAA 安全与合规架构白皮书 — 本白皮书描述了公司如何使用来 AWS 创建符合 HIPAA 标准的应用程序。
- AWS 合规资源 — 此工作簿和指南集可能适用于您所在的行业和所在地。
- AWS Config— 该 AWS 服务评估您的资源配置在多大程度上符合内部实践、行业指导方针和法规。
- AWS Security Hub— 此 AWS 服务可全面了解您的安全状态 AWS ，帮助您检查是否符合安全行业标准和最佳实践。

有关特定合规计划范围内的 AWS 服务列表，请参阅按合规计划划分的范围内的 AWS 服务。有关一般信息，请参阅 AWS 合规性计划。

Amazon Comprehend Medical 中的弹性

AWS 全球基础设施是围绕 AWS 区域和可用区构建的。 AWS 区域提供多个物理隔离和隔离的可用区，这些可用区通过低延迟、高吞吐量和高度冗余的网络相连。利用可用区，您可以设计和操作在可用区之间无中断地自动实现故障转移的应用程序和数据库。与传统的单个或多个数据中心基础架构相比，可用区具有更高的可用性、容错性和可扩展性。

有关 AWS 区域和可用区的更多信息，请参阅AWS 全球基础设施。

Amazon Comprehend Medical 中的基础设施安全性

作为一项托管服务，Amazon Comprehend Medical 受 AWS 亚马逊网络服务：安全流程概述白皮书中描述的全球网络安全程序的保护。

要通过网络访问 Comprehend Medical，您可以使用已发布的 API 调用。 AWS 客户端必须支持传输层安全性协议 (TLS) 1.0 或更高版本。建议使用 TLS 1.2 或更高版本。客户端还必须支持具有完全向前保密（PFS）的密码套件，例如 Ephemeral Diffie-Hellman（DHE）或 Elliptic Curve Ephemeral Diffie-Hellman（ECDHE）。大多数现代系统（如 Java 7 及更高版本）都支持这些模式。

此外，必须使用访问密钥 ID 和与 AWS Identity and Access Management (IAM) 委托人关联的私有访问密钥对请求进行签名。或者，您可以使用 AWS Security Token Service（AWS STS）生成临时安全凭证来对请求进行签名。

指南和配额

使用 Amazon Comprehend Medical 时，请记住以下信息。

> **Note**
>
> - Amazon Comprehend Medical 支持 UTF-8 英语（EN）字符编码。
> - Amazon Comprehend Medical 不允许在异步任务的文件路径中使用连续的正斜杠字符 `//`（）。

重要提示

Amazon Comprehend Medical 并不代替专业的医学意见、诊断或治疗。Amazon Comprehend Medical 提供置信度分数，该分数表明对检测到的实体的准确性的信心。为您的使用案例确定正确的置信度阈值，并在需要高准确度的情况下使用高置信度阈值。对于某些使用案例，结果应由经过适当培训的人力审核人员进行审核和验证。只有经过训练有素的医学专家审查结果的准确性和进行合理的医学判断后，才能在患者护理场景中使用 Amazon Comprehend Medical。

支持的区域

有关提供 Amazon Comprehend Medical 的 AWS 区域的列表，请参阅 Amazon Web Services General Reference 中的 AWS Regions and Endpoints。

节流

有关 Amazon Comprehend Medical 的限制和配额以及如何请求增加配额的更多信息，请参阅 AWS Service Quotas。

整体配额

Amazon Comprehend Medical 实时（同步）分析操作有以下限额：

资源	默认
Detectitie v2、Detitie 、Infrm 和 Infe0CM 操作的每秒事务数 (TPS)	40 TPS
Infe EDCT 操作的每秒事务数 (TPS)	2 TPS
Detectitie v2、Detitie	40000 CPS

资源	默认
itie 、Ir rm 和 Infe 0CM 操作的每秒字符数 (CPS	etectPHI xNo
Infe EDCT 操作的每秒字符数 (CPS	5000 CPS

资源	默认
DetectEntities、DetectEntities-v2 和 DetectEntities-v2 操作的最大文档大小（UTF-8 字符）	20 KB

资源	默认
Infer0-CM 和 InferRxm 操作的最大文档大小（UTF-8 字符）	10 KB
InferICD10-CM 操作的最大文档大小（UTF-8 字符）	5 KB

Amazon Comprehend Medical 批量分析（异步）操作具有以下限额：

描述	配额
StartEntitiesDetectionV2Job 、StartPHIDetectionJob 、StopEntitiesDetectionV2Job 、StopPHIDetectionJob 、StartICD10CMInferenceJob 、StartRxNormInferenceJob 、StopICD10CMInferenceJob 、StopRxNormInferenceJob 、StartSNOMEDCTnferenceJob 和 StopSNOMEDCTnferenceJob 操作的每秒事务数（TPS）	5 TPS
ListEntitiesDetectionV2Jobs 、ListPHIDetectionJobs 、DescribeEntitiesDetectionV2Job 、DescribePHIDetectionJob 、ListICD10CMInferenceJobs 、ListRxNormInferenceJobs 、DescribeICD10CMInferenceJob 、DescribeRxNormInferenceJob 、ListSNOMEDCTnferenceJobs 和 DescribeSNOMEDCTnferenceJob 操作的每秒事务数（TPS）	10 TPS
所有操作的批处理作业的最大单个文件大小	70KB
最大批处理作业大小（批处理作业中提交的所有文件的总和）	1 GB
每个操作中当前正在运行的批处理作业的最大数量	10 个作业

如果您的文本大于字符配额，请使用 segment.py 创建可供分析的较小的分段。

Amazon Comprehend Medical 文档历史记录

下表介绍了此版本的 Amazon Comprehend Medical 的文档。

变更	说明	日期
更新了 InferSNOMEDCT 的 API 版本	InferSNOMEDCT API 操作现在使用版本3.0.0.20220301 。	2024 年 6 月 26 日
更新了 API 版本 InferRxNorm	InferRxNorm API 操作现在使用版本3.0.0.20221107 。	2024 年 6 月 26 日
更新了 Infer ICD1 0CM 的 API 版本	InferICD10CM API 操作现在使用版本3.0.0.20231001 。	2024 年 6 月 26 日
更新了 DetectEntities V2 的 API 版本	DetectEntitiesV2 API 操作现在使用版本3.0.0。	2024 年 6 月 26 日
更新了 API 版本 DetectEntities	DetectEntities API 操作现在使用版本3.0.0。	2024 年 6 月 26 日
更新了 Infer ICD1 0CM 的 API 版本	API 操作 InferICD10CM 现在使用版本 3.0.0.20231001 。	2024 年 2 月 7 日
更新了 Infer ICD1 0CM 的 API 版本	API 操作 InferICD10CM 现在使用版本 2.6.0.20220401 。	2023 年 9 月 27 日
更新了 InferSNOMEDCT 的 API 版本	API 操作 InferSNOMEDCT 现在使用版本 3.0.0.20220301 。	2023 年 6 月 12 日
为 V2 添加了新的特征和属性 DetectEntities	API 操作 DetectEntitiesV2 现在使用版本	2023 年 6 月 12 日

	3.0.0。此更新添加了新的特征和属性。	
为添加了新的特征和属性 DetectEntities	API 操作 DetectEntities 现在使用版本 3.0.0。此更新添加了新的特征和属性。	2023 年 6 月 12 日
新特征已添加到 InferRxNorm	API 操作 InferRxNorm 现在使用版本 3.0.0.20221107 。此更新为检测到的药物添加了 PAST_HISTORY 特征。	2023 年 6 月 12 日
Infer 0CM 中ICD1添加了新属性	API 操作 InferICD10CM 现在使用版本 2.5.0.20220401 。此更新将 QUALITY 属性添加到医学状况类别。	2023 年 6 月 12 日
更新了 API 版本 InferRxNorm	API 操作 InferRxNorm 现在使用版本 2.1.0.20221003 。此更新提高了 InferRxNorm 模型的性能。	2023 年 4 月 28 日
更新了 InferSNOMEDCT 的 API 版本	API 操作 InferSNOMEDCT 现在使用版本 2.5.0.20220301 。此更新提高了模型的性能。	2023 年 4 月 21 日
更新了 Infer ICD1 0CM 的 API 版本	API 操作 InferICD10CM 现在使用版本 2.4.0.20220401 。此更新改进了对 DX_NAME 实体和 DIRECTION 属性的检测。	2023 年 3 月 17 日

更新了 InferSNOMEDCT 的 API 版本	API 操作 `InferSNOMEDCT` 现在使用版本 `2.4.0.20220301` 。此更新改进了 `MEDICAL_CONDITION` 和 `TEST_TREATMENT_PROCEDURE` 类别中心脏病学概念的检测。	2023 年 3 月 15 日
更新了 Infer ICD1 0CM 的 API 版本	API 操作 `InferICD10CM` 现在使用版本 `2.3.0.20220401` 。此更新改进了 `MEDICAL_CONDITION` 类别中心脏病学概念的检测。	2023 年 3 月 15 日
更新了 DetectEntities V2 的 API 版本	API 操作 `DetectEntitiesV2` 现在使用版本 `2.3.0`。此更新改进了 `MEDICAL_CONDITION` 和 `TEST_TREATMENT_PROCEDURE` 类别中心脏病学概念的检测。	2023 年 3 月 15 日
更新了 API 版本 DetectEntities	API 操作 `DetectEntities` 现在使用版本 `0.6.0`。此更新改进了 `MEDICAL_CONDITION` 和 `TEST_TREATMENT_PROCEDURE` 类别中心脏病学概念的检测。	2023 年 3 月 15 日
更新 InferSNOMEDCT 的 API 版本	API 操作 `InferSNOMEDCT` 现在使用版本 `2.3.0.20220301` 。此更新改进了使用 InfersnomedCT API 操作时对否定的检测。	2023 年 2 月 8 日

更新 Infer ICD1 0CM 的 API 版本	API 操作 `InferICD10CM` 现在使用版本 `2.2.0.20220401` 。此更新改进了使用 Infer ICD1 0CM API 操作时对否定的检测。	2023 年 2 月 8 日
更新 DetectEntities V2 的 API 版本	API 操作 `DetectEntitiesV2` 现在使用版本 `2.2.0`。此更新改进了使用 DetectEntities V2 API 操作时对否定的检测。	2023 年 2 月 8 日
更新的 API 版本 DetectEntities	API 操作 `DetectEntities` 现在使用版本 `0.5.0`。此更新改进了使用 DetectEntities API 操作时对否定的检测。	2023 年 2 月 8 日
更新 InferSNOMEDCT 的 API 版本	API 操作 `InferSNOMEDCT` 现在使用版本 `2.2.0.20220301` 。此更新修复了一个错误，该错误导致无法在 API 输出中返回已识别的全部属性集。	2022 年 12 月 9 日
更新了 InferSNOMEDCT 的 API 版本	API 操作 `InferSNOMEDCT` 现在使用版本 `2.1.0.20220301` 。此更新包括准确性提升，并检测到了新的特征。	2022 年 11 月 16 日
更新了 API 版本 InferRxNorm	API 操作 `InferRxNorm` 现在使用版本 2.1.0.20221003。此更新修复了一个错误，该错误导致无法返回已识别的全部属性集。	2022 年 11 月 16 日

更新了 Infer ICD1 0CM 的 API 版本	API 操作 `InferICD10CM` 现在使用版本 `2.1.0.202 20401` 。此更新包括准确性提升，并检测到了新的特征。	2022 年 11 月 16 日
更新了 DetectEntities V2 的 API 版本	API 操作 `DetectEnt itiesV2` 现在使用版本 `2.1.0`。此更新包括准确性提升。它还添加了新的实体、属性、关系、特征和新类别。	2022 年 11 月 16 日
更新了 API 版本 DetectEntities	API 操作 `DetectEntities` 现在使用版本 `0.4.0`。此更新包括准确性提升，并检测到了新的特征。	2022 年 11 月 16 日
更新了 `StartSNOM EDCTInferenceJob` 和 `InferSNOMEDCT` 的限制	现在，单个文件大小 `StartSNOMEDCTInfer enceJob` 和 `InferSNOM EDCT` API 操作的最大值为 5 KB。	2022 年 9 月 22 日
文本处理更新	Amazon Comprehend Medical 现在为正确解析所有文本分析 API 操作中不相关的空格和回车提供了更好的支持。	2022 年 9 月 12 日
文本处理更新	Amazon Comprehend Medical 现在为正确解析所有本体 API 操作中不相关的空格和回车提供了更好的支持。	2022 年 9 月 12 日
更新了 InferSNOMEDCT 的 API 版本	Amazon Comprehend Medical API 操作 `InferSNOWMEDCT` 现在使用版本 1.1.0.202 20301。	2022 年 7 月 8 日

更新了 Infer ICD1 0CM 的 API 版本	Amazon Comprehend Medical API 操作 `InferICD10CM` 现在使用版本 1.1.0.20220401。	2022 年 7 月 8 日
更新了可在亚马逊 Comprehend Medic InferRxNorm al 中运行的 API 版本 Amazon Medical	亚马逊 Comprehend Medical 现在为每个 RxCUI 使用 RxNorm 202 RxTerms 2-03-07 和版本。	2022 年 6 月 29 日
更新了 Amazon Comprehend Medical 的模型	Amazon Comprehend Medical 现在使用 Detect Entities V2 为 `NEGATION` 特征检测提供了更好的支持。	2022 年 4 月 25 日
Amazon Comprehend Medical 的新功能	您现在可以使用 Amazon Comprehend Medical 创建接口 VPC 端点，从而与虚拟私有云 (VPC) 建立专用连接。有关更多信息，请参阅 VPC 终端节点 (PrivateLink)。	2021 年 6 月 13 日
Amazon Comprehend Medical 的新功能	Amazon Comprehend Medical 现在为本体关联提供批量操作。这使该服务能够检测存储在 S3 存储桶中的医学文本中的实体，并将这些实体关联到标准化本体。有关更多信息，请参阅 Ontology Linking Batch Analysis。	2020 年 5 月 4 日
Amazon Comprehend Medical 的新功能	现在，您可以使用 Amazon Comprehend Medical 提取日期或时间表达式，并将其与 Amazon Comprehend Medical 当前检测到的任何实体关联起来。有关更多信息，请参阅 Detect Entities Version 2。	2020 年 3 月 2 日

新功能	Amazon Comprehend Medical 现在可以检测药物和疾病，并将这些实体与既定的国家本体和 ICD-10-联系起来。 RxNorm CMs有关更多信息，请参阅本体链接 APIs。	2019 年 12 月 16 日
新特征	Amazon Comprehend Medical 现在提供批量操作，以便您处理存储在 S3 存储桶中的医学文本。它还提供了一种可用于检查医学文本的新模型。有关更多信息，请参阅 Detect Entities Version 2。	2019 年 9 月 24 日
主要新功能	Amazon Comprehend Medical 是一项新服务，可检测非结构化临床文本中的有用信息：医生记录、出院摘要、检验结果、病例记录等。Amazon Comprehend Medical 使用自然语言处理 (NLP) 模型，利用机器学习领域的最新进展，对这些海量数据进行分类，并检索出有价值的信息，而这些信息如果没有大量的人工操作是很难检索和使用的。有关更多信息，请参阅 Amazon Comprehend Medical。	2018 年 11 月 27 日

Amazon Comprehend Medical 变更日志

以下各节详细介绍了 Amazon Comprehend Medical 服务的更新。

更新了 API 版本 DetectEntities

发布日期：2024-06-26

类型：自动

`DetectEntities`API 操作现在使用模型版本`3.0.0`。

更新了 DetectEntities V2 的 API 版本

发布日期：2024-06-26

类型：自动

`DetectEntitiesV2`API 操作现在使用模型版本`3.0.0`。

更新了 API 版本 InferRxNorm

发布日期：2024-06-26

类型：自动

`InferRxNorm`API 操作现在使用模型版本`3.0.0.20221107`。

更新了 InferSNOMEDCT 的 API 版本

发布日期：2024-06-26

类型：自动

`InferSNOMEDCT`API 操作现在使用模型版本`3.0.0.20220301`。

更新了 Infer ICD1 0CM 的 API 版本

发布日期：2024-06-26

类型：自动

`InferICD10CM`API 操作现在使用模型版本`3.0.0.20231001`。

Amazon Comprehend Medical API 操作的变更

发布日期：2024-02-07

类型：自动

在此更新中，Infer ICD1 0CM API 操作已更新。Infer ICD1 0CM API 操作现在使用模型版本 3.0.0.20231001。此更新引入了对新发布的 2024 ICD1 0CM 代码集的支持，该代码集已于 2023 年 10 月 1 日生效。此更新还提高了模型的整体准确性。

Amazon Comprehend Medical API 操作的变更

发布日期：2023 年 9 月 27 日

类型：自动

在此更新中，Infer ICD1 0CM API 操作已更新为使用模型版本 2.6.0.20220401。此更新提高了低置信度分数和模型准确性。

Amazon Comprehend Medical API 操作的变更

发布日期：2023 年 6 月 12 日

类型：自动

在此更新中，以下 API 操作现在使用新的模型版本。

- DetectEntities 现在使用模型版本 3.0.0。
- DetectEntitiesV2 现在使用模型版本 3.0.0。
- InferRxNorm 现在使用模型版本 3.0.0.20221107。
- Infer ICD1 0CM 现在使用模型版本 2.5.0.20220401。
- InfersnomedCT 现在使用模型版本 3.0.0.20220301。

此更新将 `PAST_HISTORY` 特征添加到 `DetectEntities`、`InferRxNorm` 和 `DetectEntitiesV2` API 操作中的药物类别中。

此更新还将 `QUALITY` 属性添加到 `DetectEntitiesV2`、`DetectEntities` 和 `InferICD10CM` 中的医学状况类别中。

要详细了解 Amazon Comprehend Medical 的文本分析 API 操作，请参阅 文本分析 API 操作。

要了解有关 Amazon Comprehend Medical 本体链接 API 操作的更多信息，请参阅。本体关联

Amazon Comprehend Medical API 操作的变更

发布日期：2023 年 4 月 28 日

类型：自动

在此更新中， InferRxNorm API 操作已更新。 InferRxNorm API 操作现在使用模型版本 2.1.0.20221003。此更新提高了模型的性能。

Amazon Comprehend Medical API 操作的变更

发布日期：2023 年 4 月 21 日

类型：自动

在此更新中，InferSNOMEDCT API 操作已更新。InferSNOMEDCT API 操作现在使用模型版本 2.5.0.20220301。此更新提高了模型的性能。

Amazon Comprehend Medical API 操作的变更

发布日期：2023 年 3 月 17 日

类型：自动

在此更新中，Infer ICD1 0CM API 操作已更新。此更新改进了对 `DX_NAME` 实体和 `DIRECTION` 属性的检测。它还会对系统处理进行微小的调整，从而影响特征和属性的检测以及 API 输出中的频率/置信度分数。

Amazon Comprehend Medical API 操作的变更

发布日期： 2023 年 3 月 15 日

类型：自动

在本次更新中，更新了以下 Amazon Comprehend Medical API 操作：`InferSNOMEDCT`、`InferICD10CM`、`DetectEntities` 和 `DetectEntitiesV2`。这些更新都是自动进行的，可提高对心脏病专业内的医学状况、检查、治疗和手术以及相关属性和特征的检测。

Amazon Comprehend Medical API 操作的变更

发布日期：2023 年 2 月 8 日

类型：自动

在本次更新中，更新了以下 Amazon Comprehend Medical API 操作：`InferSNOMEDCT`、`InferICD10CM`、`DetectEntities` 和 `DetectEntitiesV2`。这些更新是自动进行的，可以改善扫描临床文本时对医学状况中否定的检测。

Amazon Comprehend Medical InferSNOMEDCT API 操作的变更

发布日期： 2022 年 12 月 9 日

类型：自动

Amazon Comprehend Medical API 操作 `InferSNOMEDCT` 已更新为使用版本 `2.2.0.20220301`。此更新修复了一个错误，该错误导致无法返回已识别的全部属性集。

使用 `StartSNOMEDCTInferenceJob` API 操作执行的批量分析使用相同的 API 版本 `2.2.0.20220301`。要了解有关使用这些 API 操作的更多信息，请参阅 SNOMED CT 关联。

Amazon Comprehend Medical API 操作的变更

发布日期： 2022 年 11 月 16 日

类型：自动

在本次更新中，更新了以下 Amazon Comprehend Medical API 操作：`InferSNOMEDCT`、`InferICD10CM`、`DetectEntities` 和 `DetectEntitiesV2`。

`DetectEntities`

Amazon Comprehend Medical API 操作 `DetectEntities` 现在使用 API 版本 0.4.0。此更新包括准确性提升，并检测到了新的特征。

对于类别：`TEST_TREATMENT_PROCEDURE`，类型：`PROCEDURE_NAME`，添加了以下特征。

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

对于类别：`TEST_TREATMENT_PROCEDURE`，类型：`TEST_NAME`，添加了以下特征。

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

对于类别：`TEST_TREATMENT_PROCEDURE`，类型：`TREATMENT_NAME`，添加了以下特征。

- NEGATION

- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

对于类别：MEDICAl_CONDITION，类型：DX_NAME，添加了以下特征。

- HYPOTHETICAL
- LOW_CONFIDENCE
- PERTAINS_TO_FAMILY

DetectEntitiesV2

Amazon Comprehend Medical API 操作 DetectEntitiesV2 现在使用 API 版本 2.1.0。此更新加入了一个新类别 BEHAVIORAL_ENVIRONMENTAL_SOCIAL 并包含现有类别和特征的更新。

对于新的类别：BEHAVIORAL_ENVIRONMENTAL_SOCIAL，添加了以下类型：GENDER、RACE_ETHNICITY、ALLERGIES、TOBACCO_USE、ALCOHOL_CONSUMPTION、REC_DRU

对于类型：GENDER

- 没有可用的属性或特征。

对于类型：RACE_ETHNICITY

- 没有可用的属性或特征。

对于类型：ALLERGIES，添加了以下特征。

- NEGATION
- PAST_HISTORY

对于类型：TOBACCO_USE，添加了以下属性和特征。

- Attributes
  - AMOUNT
  - DURATION
  - FREQUENCY
- 特性
  - NEGATION
  - PAST_HISTORY

对于类型：ALCOHOL_CONSUMPTION，添加了以下属性和特征。

- Attributes
  - AMOUNT
  - DURATION
  - FREQUENCY
- 特性
  - NEGATION
  - PAST_HISTORY

对于类型：REC_DRUG_USE，添加了以下属性和特征。

- Attributes
  - AMOUNT
  - DURATION
  - FREQUENCY
- 特性
  - NEGATION
  - PAST_HISTORY

对于类型：UnmappedAttributes，添加了以下属性。

- AMOUNT

对于类别：MEDICAl_CONDITION，类型：DX_NAME，添加了以下特征。

- PERTAINS_TO_FAMILY
- HYPOTHETICAL
- LOW_CONFIDENCE

对于类别：TEST_TREATMENT_PROCEDURE，类型：PROCEDURE_NAME，添加了以下特征。

- NEGATION
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

对于类别：TEST_TREATMENT_PROCEDURE，类型：TEST_NAME，添加了以下特征。

- PAST_HISTORY
- HYPOTHETICAL

- FUTURE

对于类别：TEST_TREATMENT_PROCEDURE，类型：TREATMENT_NAME，添加了以下特征。

- NEGATION
- PAST_HISTORY
- HYPOTHETICAL
- FUTURE

使用 StartEntitiesDetectionV2Job API 操作执行的批量分析使用相同的 API 版本 2.1.0。要了解有关使用这些 API 操作的更多信息，请参阅 检测实体（版本 2）。

InferSNOMEDCT

Amazon Comprehend Medical API 操作 InferSNOMEDCT 现在使用 API 版本 2.1.0.20220301。此版本更新为以下类别和类型添加了新的特征。

对于类别：MEDICAL_CONDITION，类型：DX_NAME，添加了以下特征。

- PERTAINS_TO_FAMILY
- HYPOTHETICAL
- LOW_CONFIDENCE

对于类别：TEST_TREATMENT_PROCEDURE，类型：TEST_NAME，添加了以下特征。

- PAST_HISTORY
- FUTURE
- HYPOTHETICAL

对于类别：TEST_TREATMENT_PROCEDURE，类型：TREATMENT_NAME，添加了以下特征。

- NEGATION
- PAST_HISTORY
- FUTURE
- HYPOTHETICAL

错误修复

- 此更新修复了一个错误，该错误导致无法返回已识别的全部属性集。

使用 StartSNOMEDCTInferenceJob API 操作执行的批量分析使用相同的版本。要了解有关使用这些 API 操作的更多信息，请参阅 SNOMED CT 关联。

InferICD10CM

Amazon Comprehend Medical API 操作 `InferICD10CM` 现在使用 API 版本 2.1.0.20220401。此版本更新为以下类别和类型添加了新的特征。

对于类别：`MEDICAL_CONDITION`，类型：`DX_NAME`，添加了以下特征。

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

错误修复

- 此更新修复了一个错误，该错误导致无法返回已识别的全部属性集。

使用 `StartRxNormInferenceJob` API 操作执行的批量分析使用相同的版本。要了解有关使用 `InferRxNorm` 和 `StartRxNormInferenceJob` API 的更多信息，请参阅 RxNorm 链接。

InferRxNorm

亚马逊 Comprehend Medical API 操作现在使用 API 版本 InferRxNorm 。`1.3.1.20221003`

错误修复

- 此更新修复了一个错误，该错误导致无法返回已识别的全部属性集。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。