开发人员指南

Amazon Translate

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

Amazon Translate: 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 Translate？ 1
使用案例 1
新用户 2
Amazon Translate 定价 2
Amazon Translate API 参考 2
支持的语言 3
支持的语言 3
Amazon Translate 功能支持的语言 7
工作方式 8
输入内容支持的格式 8
自定义您的翻译 8
自动语言检测 9
异常处理 9
设置 10
注册获取 AWS 账户 10
创建具有管理访问权限的用户 10
设置 AWS CLI 12
授予编程式访问权限 12
与... 合作 AWS SDKs 13
入门 15
开始使用（控制台） 15
入门 (AWS CLI) 17
使用命令行翻译文本 17
后续步骤 17
开始使用（软件开发工具包） 17
使用适用于 Java 的 SDK 18
使用适用于 Python 的 AWS SDK 18
其他 SDK 示例 18
翻译处理模式 19
实时翻译 19
实时翻译（控制台） 19
实时翻译 (API) 23
异步批处理 26
区域可用性 26

先决条件 27
运行作业 32
监控和分析 38
获取结果 40
自定义您的翻译 43
使用 do-not-translate标签 43
在控制 do-not-translate台上使用 43
do-not-translate与 API 一起使用 44
使用自定义术语进行自定义 44
创建自定义术语 45
使用自定义术语 47
使用 Python 开发工具包的示例 48
加密术语 50
最佳实践 51
简洁明了 52
使用简洁设置 52
支持的语言 52
掩盖亵渎言论 53
使用脏话设置 54
语言不受支持 54
设定形式 55
使用正式性设置 55
支持的语言 56
使用并行数据进行自定义 56
区域可用性 58
Amazon Translate 的并行数据输入文件 58
添加并行数据 63
查看和管理并行数据 66
代码示例 71
基本功能 71
操作 72
场景 90
构建 Amazon Transcribe 流式传输应用程序 90
构建 Amazon Lex 聊天机器人 91
构建 Amazon SNS 应用程序 92
创建用于分析客户反馈的应用程序 93

开始使用翻译任务 99
标记 102
标记新资源 103
查看、编辑和删除标签 104
安全性 106
数据保护 106
静态加密 107
传输中加密 108
身份和访问管理 108
受众 109
使用身份进行身份验证 109
使用策略管理访问 112
Amazon Translate 如何与 IAM 配合使用 114
基于身份的策略示例 120
AWS 托管式策略 126
故障排除 128
监控 129
使用监控 CloudWatch 132
使用记录 Amazon Translate API 调用 AWS CloudTrail 133
CloudWatch Amazon Translate 的指标和维度 136
使用监控 EventBridge 137
合规性验证 140
恢复能力 141
基础结构安全性 141
VPC 端点（AWS PrivateLink） 141
Amazon Translate VPC 端点注意事项 142
为 Amazon Translate 创建接口 VPC 端点 142
为 Amazon Translate 创建 VPC 端点策略 142
准则和限额 144
支持的 AWS 区域 144
合规 144
节流 144
指南 144
服务限额 145
文档历史记录 148
API 参考 158

AWS 词汇表 159
..... clx

什么是 Amazon Translate？

Amazon Translate 是一种文本翻译服务，它使用先进的机器学习技术，按需提供高质量的翻译。您可以使用 Amazon Translate 来翻译非结构化文本文档或构建使用多种语言的应用程序。有关 Amazon Translate 所支持语言的信息，请参阅 支持的语言和语言代码。

主题

- 使用案例
- 您是首次使用 Amazon Translate 的用户吗？
- Amazon Translate 定价
- Amazon Translate API 参考

使用案例

请使用 Amazon Translate 执行以下操作：

通过集成 Amazon Translate 在您的应用程序中实现多语言用户体验：

- 翻译公司撰写的内容，如会议纪录、技术报告、知识库文章、帖子等。
- 翻译人际通讯内容，如电子邮件、游戏内聊天、客户服务聊天，以便客户和员工能够通过其首选语言进行联系。

处理和管理贵公司的传入数据：

- 以多种语言分析文本，如社交媒体和新闻推送。
- 以多种语言搜索信息，如 eDiscovery 案例信息。

通过将 Amazon Translate 与其他 AWS 服务集成，以实现语言无关的处理：

- 使用 Amazon Comprehend，从社交媒体流等非结构化文本中提取指定的实体、情绪和关键短语。
- 使用 Amazon Transcribe，以多种语言提供字幕和实时字幕。
- 使用 Amazon Polly 以朗读翻译的内容。
- 翻译 Amazon S3 中存储的文档存储库。
- 翻译存储在以下数据库中的文本：Amazon DynamoDB、Amazon Aurora 和 Amazon Redshift。
- 将工作流与 AWS Lambda 或 AWS Glue 无缝集成。

您是首次使用 Amazon Translate 的用户吗？

如果您是首次使用 Amazon Translate 的用户，我们建议您从以下主题开始：

1. Amazon Translate 的工作原理 – Amazon Translate 简介
2. 开始使用 Amazon Translate – 介绍如何设置您的 AWS 账户和开始使用 Amazon Translate。
3. 使用 Amazon Translate 的代码示例 AWS SDKs— 使用代码示例来探索 Amazon Translate APIs。

您还可以使用以下资源来了解 Amazon Translate 服务：

- AWS 机器学习博客中包含有关 Amazon Translate 的实用文章。
- Amazon Translate 深入探讨视频系列提供有关 Amazon Translate 的介绍视频。

Amazon Translate 定价

与其他 AWS 产品一样，使用 Amazon Translate 没有合同或最低承诺。有关使用 Amazon Translate 的成本的更多信息，请参阅 Amazon Translate 定价。

Amazon Translate API 参考

Amazon Translate API 参考现在是一个单独的文档。有关更多信息，请参阅 Amazon Translate API 参考。

支持的语言和语言代码

Amazon Translate 提供源语言（输入语言）和目标语言（输出语言）之间的翻译。源语言-目标语言组合称为语言对。

> **Note**
>
> 如果您将源语言和目标语言指定为相同的语言，Amazon Translate 不会向您收取翻译费用。如果您将源语言设置为 auto，则可能需要为自动检测付费。有关更多信息，请参阅 自动语言检测。

主题

- 支持的语言
- Amazon Translate 功能支持的语言

支持的语言

Amazon Translate 支持在下表中列出的语言之间翻译文本。语言代码列使用 ISO 639-1 两位语言代码。对于语言的国家/地区变体，该表遵循 RFC 5646 格式，即在后面加一个破折号，然后加上 ISO 3166 两位国家/地区代码。例如，墨西哥西班牙语变体的语言代码是 `es-MX`。

语言	语言代码
南非荷兰语	af
阿尔巴尼亚语	sq
阿姆哈拉语	am
阿拉伯语	ar
亚美尼亚语	hy
阿塞拜疆语	az
孟加拉语	bn

语言	语言代码
波斯尼亚语	bs
保加利亚语	bg
加泰罗尼亚语	ca
中文（简体）	zh
中文（繁体）	zh-TW
克罗地亚语	hr
捷克语	cs
丹麦语	da
达里语	fa-AF
荷兰语	nl
English	en
爱沙尼亚语	et
波斯语	fa
菲律宾语、塔加洛语	tl
芬兰语	fi
法语	fr
法语（加拿大）	fr-CA
格鲁吉亚语	ka
德语	de
希腊语	el

语言	语言代码
古吉拉特语	gu
海地克里奥尔语	ht
豪萨语	ha
希伯来语	he
印地语	hi
匈牙利语	hu
冰岛语	is
印度尼西亚语	id
爱尔兰语	ga
意大利语	it
日语	ja
卡纳达语	kn
哈萨克语	kk
韩语	ko
拉脱维亚语	lv
立陶宛语	lt
马其顿语	mk
马来语	ms
马来亚拉姆语	ml
马耳他语	mt

语言	语言代码
马拉地语	mr
蒙古语	mn
挪威语（布克莫尔语）	no
普什图语	ps
波兰语	pl
葡萄牙语（巴西）	pt
葡萄牙语（葡萄牙）	pt-PT
旁遮普语	pa
罗马尼亚语	ro
俄语	ru
塞尔维亚语	sr
僧伽罗语	si
斯洛伐克语	sk
斯洛文尼亚语	sl
索马里语	so
西班牙语	es
西班牙语（墨西哥）	es-MX
斯瓦西里语	sw
瑞典语	sv
泰米尔语	ta

语言	语言代码
泰卢固语	te
泰语	th
土耳其语	tr
乌克兰语	uk
乌尔都语	ur
乌兹别克斯坦语	uz
越南语	vi
威尔士语	cy

Amazon Translate 功能支持的语言

以下各节介绍了 Amazon Translate 功能支持的语言。

- 实时文档翻译 – 支持从英语翻译成任何支持的语言，以及从任何支持的语言翻译成英语。有关实时翻译的详细信息，请参阅实时翻译。
- 简洁 — 有关此功能支持的语言，请参阅在 Amazon Translate 中使用简洁性。
- 脏话掩蔽 – 有关此特征支持的语言，请参阅在 Amazon Translate 中掩蔽污言秽语。
- 正式性 – 有关此特征支持的语言，请参阅 在 Amazon Translate 中设置正式性。

Amazon Translate 的工作原理

使用 Amazon Translate 服务将内容从源语言（输入内容的语言）翻译成目标语言（您为翻译输出选择的语言）。在批处理作业中，您可以将文件从一种或多种源语言翻译成一种或多种目标语言。有关支持的语言的更多信息，请参阅支持的语言和语言代码。

主题

- 输入内容支持的格式
- 自定义您的翻译
- 自动语言检测
- 异常处理

输入内容支持的格式

Amazon Translate 支持以下格式的输入内容：

- 对于实时翻译：
  - 输入文本-UTF-8 格式的纯文本。Amazon Translate 以 UTF-8 文本的形式提供输出内容。
  - 一个输入文件 — 包含纯文本 (.txt)、HTML (.html) 或 Word (.docx) 内容的文件。Amazon Translate 以与输入文件格式相同的文件形式提供输出内容。
- 对于批量翻译作业：
  - 输入文件集合-您上传到 Amazon S3 位置的一个或多个文件。支持的文件格式包括纯文本 (.txt)、HTML (.html)、Word (.docx)、Excel (.xlsx)、(.pptx) 和 XLIFF 1.2 PowerPoint (.xlf)。Amazon Translate 以文件形式提供输出内容。每个输出文件的文件格式与输入文件格式相匹配。

自定义您的翻译

您可以使用以下功能自定义您通过 Amazon Translate 制作的翻译：

- Do-not-translate 标签 — 使用开始和结束标签来指定您不想翻译的内容（在 HTML 内容中）。
- 自定义术语 – 定义您希望 Amazon Translate 如何翻译品牌名称等特定术语。
- 简洁 — 缩短大多数翻译的翻译输出的长度（与不简洁的翻译输出相比）。实时文本翻译支持简洁性。

- 亵渎 — 掩盖翻译输出中的亵渎单词和短语。
- 形式化 — 设置翻译输出中的语言形式化级别。
- 并行数据-调整翻译输出以反映您提供的示例翻译样本中的风格、语气和单词选择。

有关信息，请参阅自定义翻译。

自动语言检测

Amazon Translate 可以自动检测源文本中使用的语言。要使用自动语言检测，请指定 `auto` 作为源语言。Amazon Translate 将代表您调用 Amazon Comprehend，以确定源文本中使用的语言。一旦选择自动语言检测，即表示您同意 Amazon Comprehend 服务条款和协议。有关 Amazon Comprehend 定价的信息，请参阅 Amazon Comprehend 定价。

异常处理

如果您指定的源语言或目标语言不受支持，Amazon Translate 将返回以下异常之一：

- UnsupportedLanguagePairException— Amazon Translate 支持所有支持的语言之间的翻译。如果源语言或目标语言不受支持，则返回此异常。有关更多信息，请参阅支持的语言。
- DetectedLanguageLowConfidenceException— 如果您使用自动语言检测，而 Amazon Translate 对检测到正确源语言的置信度不高，则会返回此异常。如果可接受低信心级别，则可以使用异常中返回的源语言。

设置

首次使用 Amazon Translate 前，请完成以下任务：

设置任务

- 注册获取 AWS 账户
- 创建具有管理访问权限的用户
- 安装和配置 AWS Command Line Interface (AWS CLI)
- 授予编程式访问权限
- 将此服务与 AWS SDK 配合使用

注册获取 AWS 账户

如果您没有 AWS 账户，请完成以下步骤来创建一个。

要注册 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。

   注册过程的一部分涉及接听电话或短信，并在电话键盘上输入验证码。

   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。

AWS 注册过程完成后会向您发送一封确认电子邮件。您可以随时前往 https://aws.amazon.com/并选择“我的账户”，查看您当前的账户活动并管理您的账户。

创建具有管理访问权限的用户

注册后，请保护您的安全 AWS 账户 AWS 账户根用户 AWS IAM Identity Center，启用并创建管理用户，这样您就不会使用 root 用户执行日常任务。

保护你的 AWS 账户根用户

1. 选择 Root 用户并输入您的 AWS 账户 电子邮件地址，以账户所有者的身份登录。AWS Management Console在下一页上，输入您的密码。

   要获取使用根用户登录方面的帮助，请参阅《AWS 登录 用户指南》中的 Signing in as the root user。

2. 为您的根用户启用多重身份验证（MFA）。

   有关说明，请参阅 I A M 用户指南中的为 AWS 账户 根用户启用虚拟 MFA 设备（控制台）。

创建具有管理访问权限的用户

1. 启用 IAM Identity Center。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的 Enabling AWS IAM Identity Center。

2. 在 IAM Identity Center 中，为用户授予管理访问权限。

   有关使用 IAM Identity Center 目录 作为身份源的教程，请参阅《用户指南》 IAM Identity Center 目录中的使用默认设置配置AWS IAM Identity Center 用户访问权限。

以具有管理访问权限的用户身份登录

- 要使用您的 IAM Identity Center 用户身份登录，请使用您在创建 IAM Identity Center 用户时发送到您的电子邮件地址的登录网址。

  有关使用 IAM Identity Center 用户登录的帮助，请参阅AWS 登录 用户指南中的登录 AWS 访问门户。

将访问权限分配给其他用户

1. 在 IAM Identity Center 中，创建一个权限集，该权限集遵循应用最低权限的最佳做法。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的 Create a permission set。

2. 将用户分配到一个组，然后为该组分配单点登录访问权限。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的 Add groups。

安装和配置 AWS Command Line Interface (AWS CLI)

您可以使用与 Amazon T AWS CLI ranslate 进行交互式通话。

要安装和配置 AWS CLI

1. 安装 AWS CLI。有关说明，请参阅《AWS Command Line Interface 用户指南》中的以下主题：

   安装或更新最新版本的 AWS Command Line Interface

2. 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：

   配置 AWS Command Line Interface

授予编程式访问权限

如果用户想在 AWS 外部进行交互，则需要编程访问权限 AWS Management Console。授予编程访问权限的方式取决于正在访问的用户类型 AWS。

要向用户授予编程式访问权限，请选择以下选项之一。

哪个用户需要编程式访问权限？	目的	方式
人力身份 （在 IAM Identity Center 中管理的用户）	使用临时证书签署向 AWS CLI AWS SDKs、或发出的编程请求 AWS APIs。	按照您希望使用的界面的说明进行操作。 • 有关的 AWS CLI，请参阅《AWS Command Line Interface 用户指南》 AWS IAM Identity Center中的配置 AWS CLI 以使用。 • 有关工具和 AWS SDKs AWS APIs，请参阅《工具参考指南》中的 IAM 身份中心身份验证AWSSDKs 和工具参考指南。

哪个用户需要编程式访问权限？	目的	方式
IAM	使用临时证书签署向 AWS CLI AWS SDKs、或发出的编程请求 AWS APIs。	按照 IAM 用户指南中的将临时证书与 AWS 资源配合使用中的说明进行操作。
IAM	（不推荐使用） 使用长期凭证签署向 AWS CLI AWS SDKs、或发出的编程请求 AWS APIs。	按照您希望使用的界面的说明进行操作。 • 有关信息 AWS CLI，请参阅用户指南中的使用 IAM 用户证书进行身份验证。AWS Command Line Interface • 有关 AWS SDKs 和工具，请参阅《工具参考指南》AWS SDKs 和《工具参考指南》中的使用长期凭证进行身份验证。 • 有关信息 AWS APIs，请参阅 IAM 用户指南中的管理 IAM 用户的访问密钥。

将此服务与 AWS SDK 配合使用

AWS 软件开发套件 (SDKs) 可用于许多流行的编程语言。每个软件开发工具包都提供 API、代码示例和文档，使开发人员能够更轻松地以其首选语言构建应用程序。

SDK 文档	代码示例
适用于 C++ 的 AWS SDK	适用于 C++ 的 AWS SDK 代码示例
AWS CLI	AWS CLI 代码示例
适用于 Go 的 AWS SDK	适用于 Go 的 AWS SDK 代码示例
适用于 Java 的 AWS SDK	适用于 Java 的 AWS SDK 代码示例

SDK 文档	代码示例
适用于 JavaScript 的 AWS SDK	适用于 JavaScript 的 AWS SDK 代码示例
适用于 Kotlin 的 AWS SDK	适用于 Kotlin 的 AWS SDK 代码示例
适用于 .NET 的 AWS SDK	适用于 .NET 的 AWS SDK 代码示例
适用于 PHP 的 AWS SDK	适用于 PHP 的 AWS SDK 代码示例
AWS Tools for PowerShell	PowerShell 代码示例工具
适用于 Python (Boto3) 的 AWS SDK	适用于 Python (Boto3) 的 AWS SDK 代码示例
适用于 Ruby 的 AWS SDK	适用于 Ruby 的 AWS SDK 代码示例
AWS SDK for Rust	AWS SDK for Rust 代码示例
适用于 SAP ABAP 的 AWS SDK	适用于 SAP ABAP 的 AWS SDK 代码示例
AWS SDK for Swift	AWS SDK for Swift 代码示例

示例可用性

找不到所需的内容？ 通过使用此页面底部的提供反馈链接请求代码示例。

开始使用 Amazon Translate

Amazon Translate 入门最简单的方法是使用控制台翻译一些文本。您也可通过命令行尝试进行 API 操作。您也可以安装其中一个 AWS SDKs 来使用 Amazon Translate API 操作。

主题

- 开始使用（控制台）
- 入门 (AWS CLI)
- 开始使用（软件开发工具包）

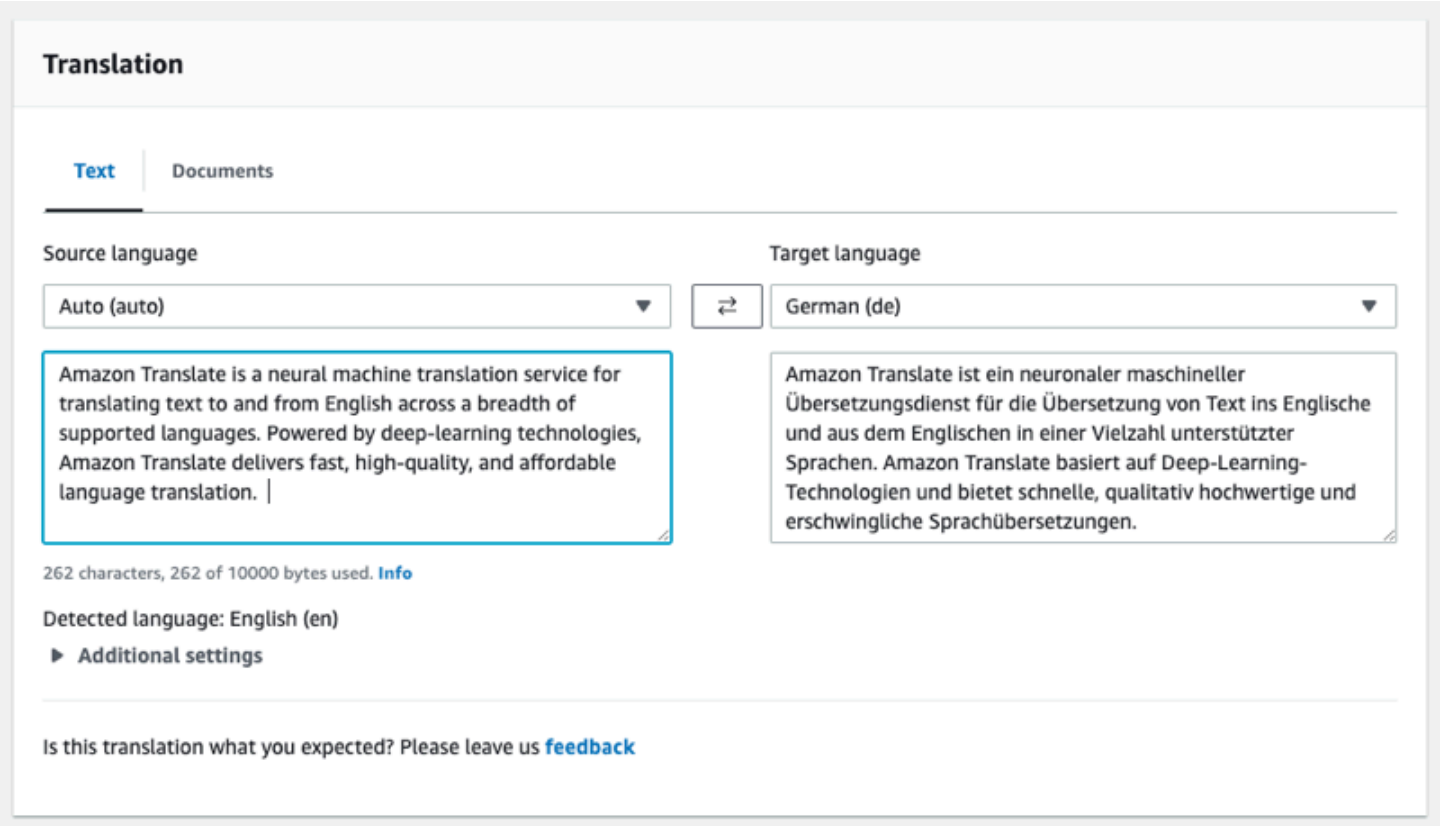
开始使用（控制台）

Amazon Translate 入门最简单的方法是使用控制台翻译一些文本。您可以使用控制台翻译最多 10,000 字节的文本。如果您尚未查看 Amazon Translate 的工作原理 中的概念和术语，我们建议您在继续之前进行查看。

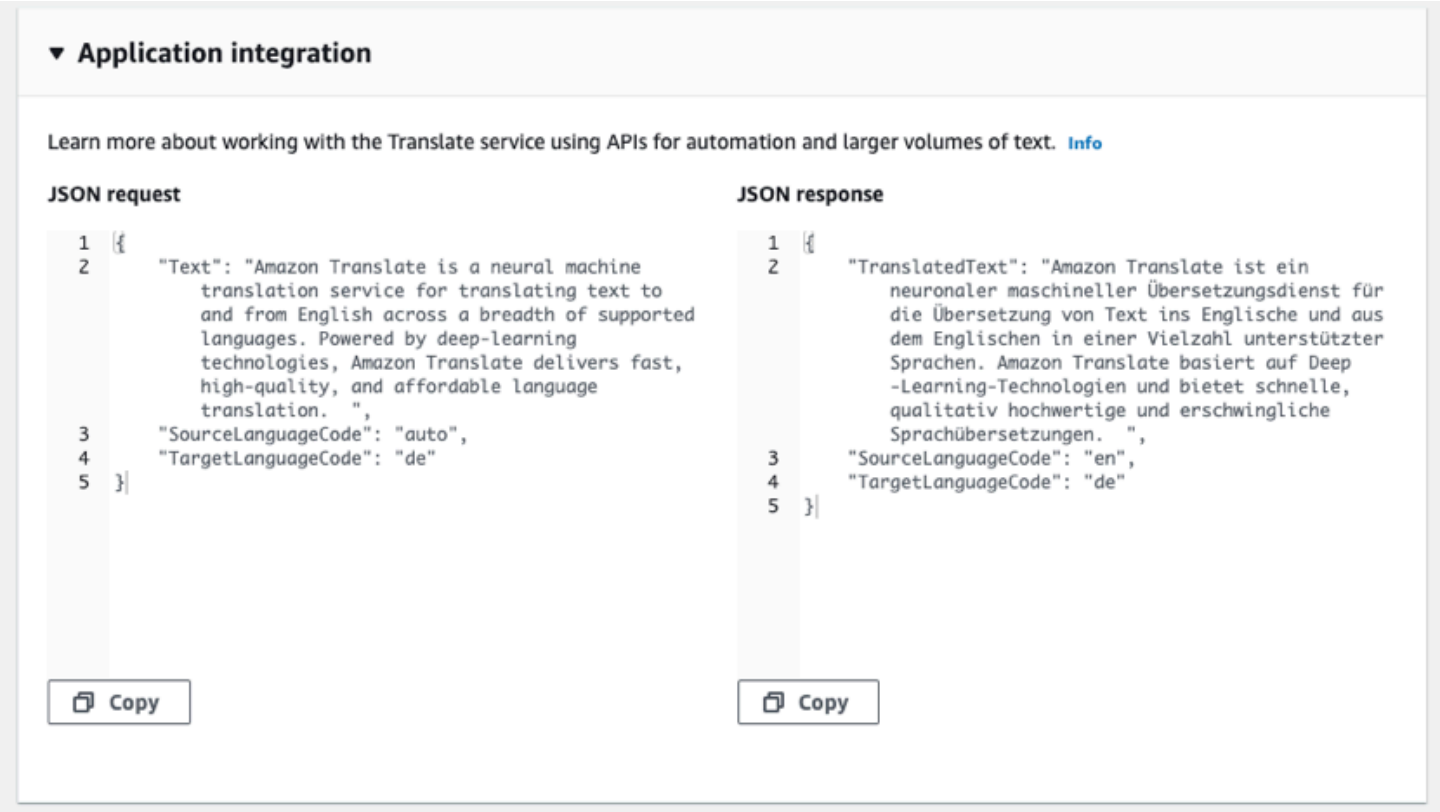
打开 Amazon Translate 控制台。

如果这是您首次使用 Amazon Translate，请选择启动实时翻译。

在实时翻译中，选择源语言和目标语言。Amazon Translate 会自动检测源语言，或者你可以选择一种源语言。在左侧文本框中输入您要翻译的文本。翻译后的文本将显示在右侧文本框中。



在应用程序集成部分，您可以看到TranslateText操作的 JSON 输入和输出。



入门 (AWS CLI)

在下面的练习中，您将使用 AWS 命令行界面 (AWS CLI) 翻译文本。要完成该练习，您需要熟悉 CLI 并且具有文本编辑器。有关更多信息，请参阅 安装和配置 AWS Command Line Interface (AWS CLI)。

要通过命令行使用 Amazon Translate，您需要在支持 Amazon Translate 服务的区域运行该命令。有关可用的端点和区域的列表，请参阅《AWS 一般参考》中的 Amazon Translate 区域和端点。

使用命令行翻译文本

以下示例演示如何从命令行使用 TranslateText 操作来翻译文本。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 (^)。在命令行处，键入以下内容。

```
aws translate translate-text \
            --region region \
            --source-language-code "en" \
            --target-language-code "es" \
            --text "hello, world"
```

响应将为以下 JSON：

```
{
    "TargetLanguageCode": "es",
    "Text": "Hola, mundo",
    "SourceLanguageCode": "en"
}
```

后续步骤

要了解使用 Amazon Translate 的其他方式，请参阅使用 Amazon Translate 的代码示例 AWS SDKs。

开始使用（软件开发工具包）

AWS SDKs 提供各种计算机语言。开发工具包管理许多您客户端的 API 连接详细信息，比如计算签名、处理请求重试和处理错误。有关更多信息，请参阅 AWS SDKs。

以下示例演示如何使用 Java 和 Python 使用 Amazon Translate TranslateText操作。使用 SDKs 来了解 Amazon Translate API，也可以将其用作您自己的应用程序的构建块。

主题

- 使用翻译文本 适用于 Java 的 AWS SDK
- 使用翻译文本 AWS SDK for Python (Boto)
- 其他 SDK 示例

使用翻译文本 适用于 Java 的 AWS SDK

AWS 提供了如何在 Java 中使用该TranslateText操作的GitHub 示例。要运行此示例，你需要 适用于 Java 的 AWS SDK。有关安装适用于 Java 的软件开发工具包的说明，请参阅设置适用于 Java 的 AWS 开发工具包 2.x。

使用翻译文本 AWS SDK for Python (Boto)

以下示例说明如何在 Python 中使用该TranslateText操作。要运行此示例，请通过安装 Python 软件开发工具包 AWS CLI。有关说明，请参阅 the section called “设置 AWS CLI”。

```
import boto3

translate = boto3.client(service_name='translate', region_name='region', use_ssl=True)

result = translate.translate_text(Text="Hello, World",
            SourceLanguageCode="en", TargetLanguageCode="de")
print('TranslatedText: ' + result.get('TranslatedText'))
print('SourceLanguageCode: ' + result.get('SourceLanguageCode'))
print('TargetLanguageCode: ' + result.get('TargetLanguageCode'))
```

有关支持的语言代码的列表，请参阅 支持的语言和语言代码

其他 SDK 示例

使用 Amazon Translate 的代码示例 AWS SDKs有关使用.NET 和 SAP ABAP 的示例，请参阅。

翻译处理模式

翻译文档时，您可以使用两种不同的翻译处理模式：实时翻译或异步批处理。您使用的模式取决于目标文档的大小和类型，并会影响您提交翻译作业和查看其结果的方式。

- 实时翻译 – 您做出同步翻译少量文本（或文本文件）的请求，Amazon Translate 会立即回复翻译后的文本。
- 异步批处理 — 您将一组文档放在亚马逊简单存储服务 (Amazon S3) 的某个位置，然后启动异步处理任务来翻译这些文档。Amazon Translate 将翻译后的输出文档发送到指定的 Amazon S3 地点。

实时翻译

Amazon Translate 提供实时文档和文本翻译操作，可立即返回翻译。您可以使用控制台或 API 执行实时翻译。

主题

- 使用控制台进行实时翻译
- 使用 API 进行实时翻译

使用控制台进行实时翻译

要使用控制台进行实时翻译，请将输入文本粘贴到源语言文本框中，或者以文件形式提供输入文本。或者，您可以设置所需的形式级别、亵渎掩盖、简洁和自定义术语等功能。

您可以将自动语言检测与实时翻译配合使用，但可能需要付费。有关更多信息，请参阅 自动语言检测。

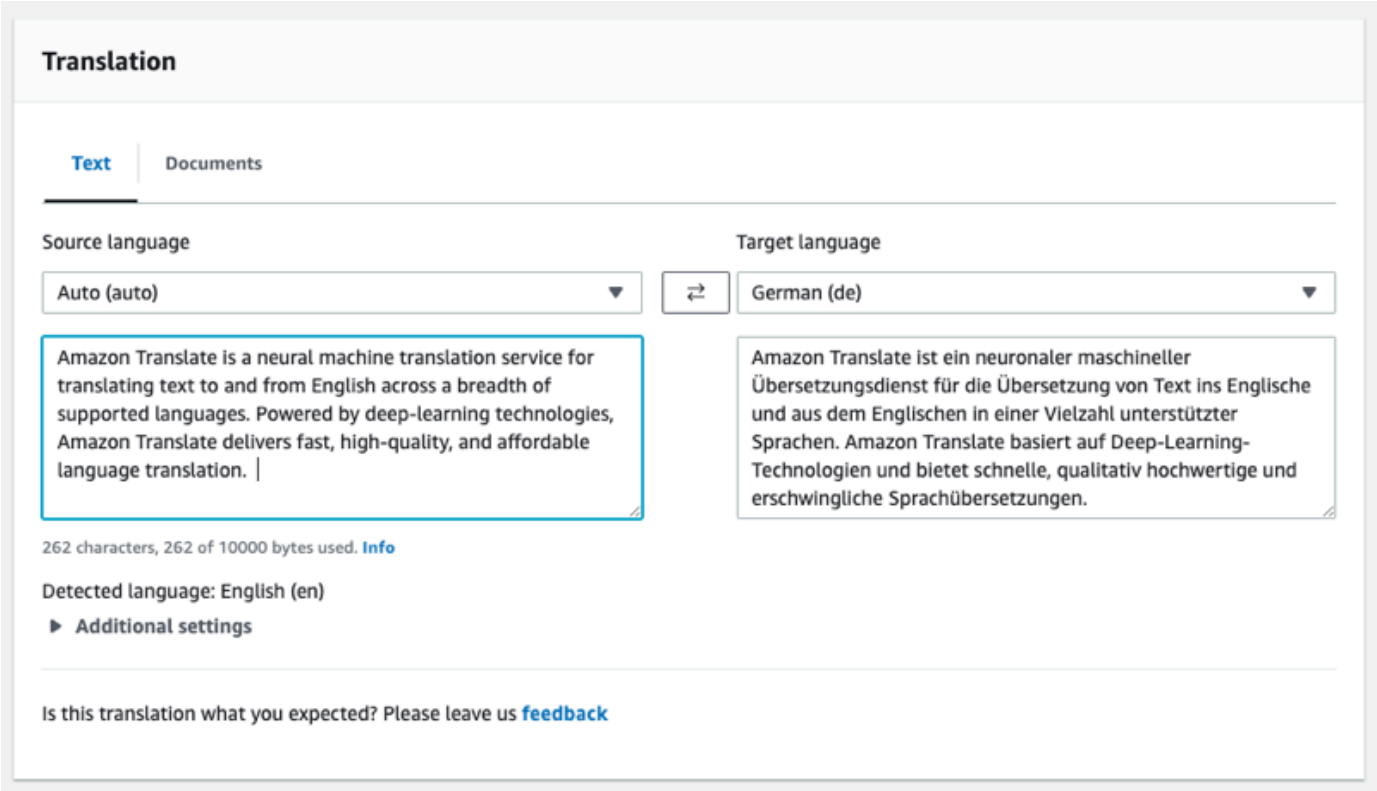
主题

- 翻译文本
- 翻译文档
- 查看等效的 API 请求和响应数据
- 使用翻译特征

翻译文本

使用 Amazon Translate 控制台翻译多达 10000 字节的文本。

1. 打开 Amazon Translate 控制台。
2. 在左侧导航菜单中，选择实时翻译。
3. 在源语言中，选择源文本的语言，或者将该值保留为自动以自动进行检测。
4. 在目标语言中，选择一种语言。
5. 在源语言文本框中输入或粘贴文本。控制台在目标语言文本框中显示翻译后的文本。

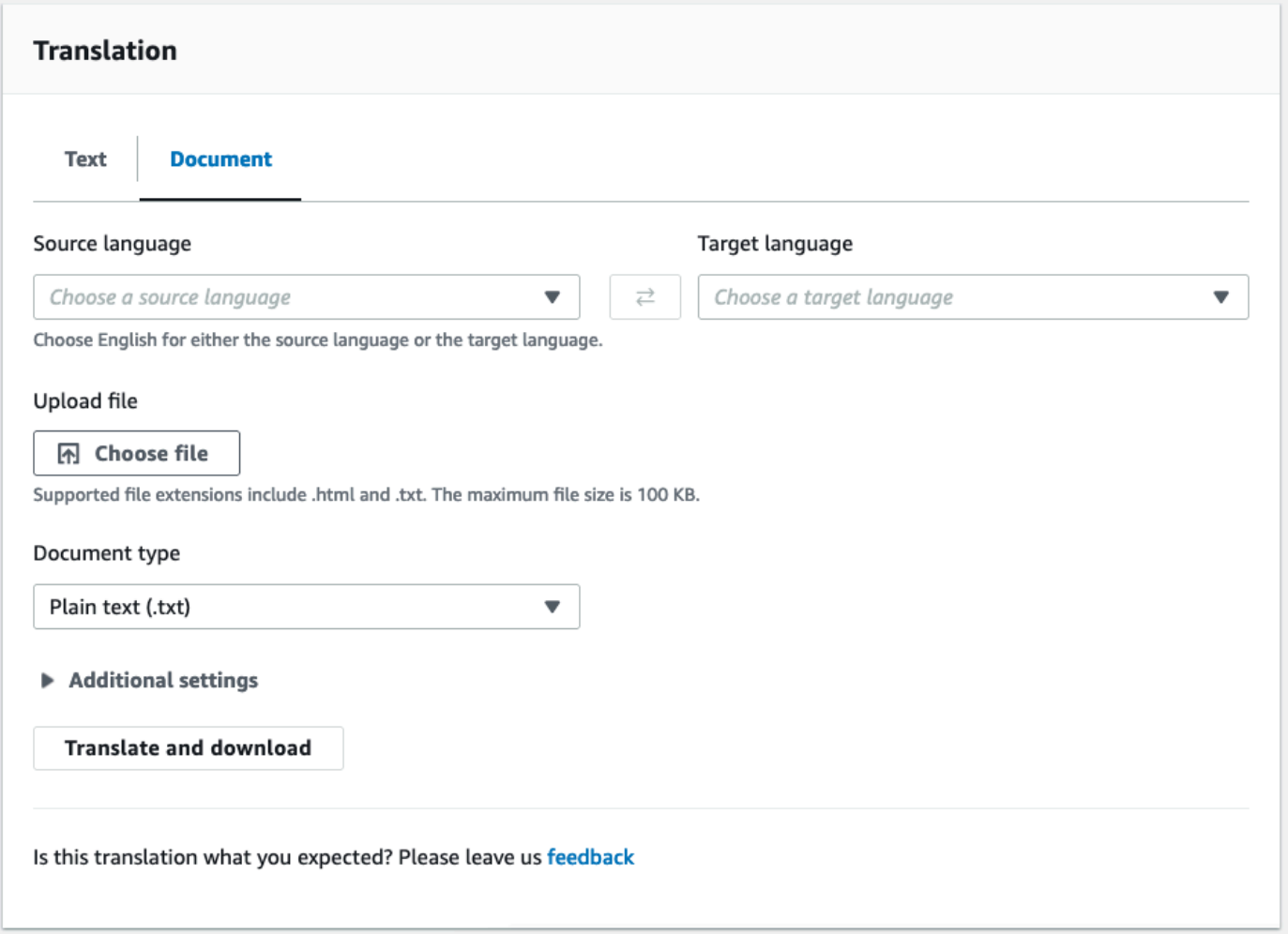


翻译文档

实时文档翻译支持从英语翻译成任何支持的语言，以及从任何支持的语言翻译成英语。

要使用 Amazon Translate 控制台翻译文档，请执行以下操作：

1. 打开 Amazon Translate 控制台。
2. 在左侧导航菜单中，选择实时翻译。
3. 在翻译面板中，选择文档选项卡。



4. 在源语言中，选择源文本的语言，或选择自动以自动进行检测。
5. 在目标语言中，选择一种语言。如果源语言不是英语，您必须选择英语作为目标语言。
6. 在上传文件下，选择选择文件，然后输入源文件的路径。最大文件大小为 100 KB。
7. 在文档类型中，选择翻译源文件的格式。文档翻译支持纯文本、HTML 或 Word (.docx) 输入文件。
8. 选择翻译。

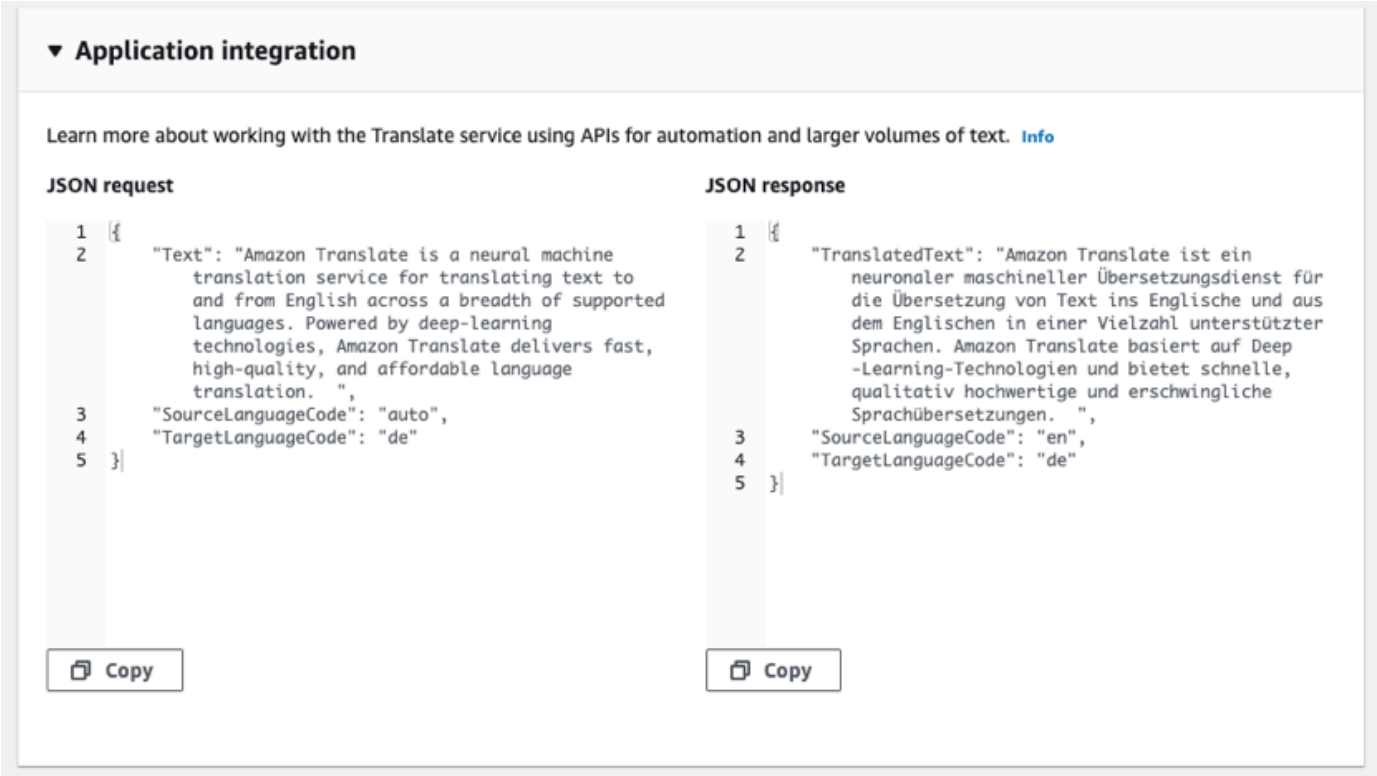
   翻译任务完成后，选择下载翻译，将翻译后的文档下载到本地硬盘。翻译后的文档格式（文本、HTML 或 Word）与输入的文档相同。

查看等效的 API 请求和响应数据

使用控制台翻译输入文本或文档后，您可以查看 JSON 格式的等效 API 请求数据和响应数据。

1. 在翻译面板下方，展开应用程序集成面板。

控制台以 JSON 格式显示等效翻译请求数据。



2. 您可以复制 JSON 请求以在TranslateText或 TranslateDocumentAPI 操作中使用。
3. JSON 响应面板中的 JSON 输出与 API 生成的输出相匹配。

使用翻译特征

要在 Amazon Translate 控制台中使用翻译特征，请执行以下操作：

1. 打开 Amazon Translate 控制台。
2. 在左侧导航菜单中，选择实时翻译。
3. 如前面的步骤所述，提供源语言、目标语言和输入数据（文本或文档）。
4. 在其他设置下，您可以选择使用以下设置自定义翻译作业的输出：

   自定义术语

   选择自定义术语文件。如果文件的输入文本中包含源术语条目，则 Amazon Translate 将使用术语文件中的译文。

   有关更多信息，请参阅 使用自定义术语自定义您的翻译。

简洁

缩短大多数翻译的翻译输出的长度（与不简洁的翻译输出相比）。Amazon Translate 支持简洁翻译文本，但不支持翻译文档。为简洁起见，如果源语言和目标语言构成不支持的语言对，Amazon Translate 将忽略简洁设置。

有关支持的语言的信息，请参阅在 Amazon Translate 中使用简洁性。

脏话

掩蔽翻译输出中的污言秽语。Amazon Translate 并不是在所有受支持的语言中都支持掩盖脏话。

有关更多信息，请参阅 在 Amazon Translate 中掩蔽污言秽语。

正式性

对于某些目标语言，您可以将正式性设置为正式或非正式。如果目标语言不支持正式性设置，Amazon Translate 将忽略该设置。

有关更多信息，请参阅 在 Amazon Translate 中设置正式性。

5. 对于文档翻译，请选择 Translate 以使用所选特征翻译文档。

对于文本翻译，在您选择每项特征后，控制台会将翻译特征应用于翻译后的文本。

使用 API 进行实时翻译

Amazon Translate 提供以下实时翻译操作来支持交互式应用程序：

- TranslateText— 翻译一段文本。
- TranslateDocument— 翻译文件的内容（纯文本、HTML 或.docx）。

这些同步操作会直接将翻译结果返回到您的应用程序。如果您在这些操作中使用自动语言检测，则可能会产生费用。有关更多信息，请参阅 自动语言检测。

翻译文本

使用该TranslateText操作翻译单个文本块。

使用命令行翻译文本

以下示例说明如何通过命令行使用该TranslateText操作。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 (^)。

在命令行输入以下命令。

```
aws translate translate-text \
            --region region \
            --source-language-code "en" \
            --target-language-code "es" \
            --text "hello, world"
```

命令使用以下 JSON 进行响应：

```
{
    "TargetLanguageCode": "es",
    "TranslatedText": "Hola, mundo",
    "SourceLanguageCode": "en"
}
```

使用 JSON 文件翻译文本

此示例演示如何使用 JSON 文件翻译更长的文本块。您可以在命令行上指定源语言和目标语言，或在 JSON 文件中指定。

> **Note**
>
> 示例 JSON 文件已设置为便于阅读的格式。重新设置 `"Text"` 字段的格式以删除换行符。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 (^)。

使用 JSON 文件翻译文本

1. 将以下文本复制到名为 `translate.json` 的 JSON 文件中：

```
{
    "Text": "Amazon Translate translates documents between languages in
    real time. It uses advanced machine learning technologies
    to provide high-quality real-time translation. Use it to
```

```
    translate documents or to build applications that work in
    multiple languages.",
    "SourceLanguageCode": "en",
    "TargetLanguageCode": "fr"
}
```

2. 在中 AWS CLI，运行以下命令：

```
aws translate translate-text \
            --region region \
            --cli-input-json file://translate.json > translated.json
```

此命令将输出包含以下 JSON 文本的 JSON 文件：

```
{
    "TargetLanguageCode": "fr",
    "TranslatedText": "Amazon Translate traduit les documents entre
    les langue en temps réel. Il utilise des technologies
    avancées d'apprentissage de la machine pour fournir
    une traduction en temps réel de haute qualité. Utilisez-le
    pour traduire des documents ou pour créer des applications
    qui fonctionnent en plusieurs langues.",
    "SourceLanguageCode": "en"
}
```

翻译文档

使用该TranslateDocument操作翻译文本、HTML 或 Word (.docx) 文档，并将翻译结果直接返回到您的应用程序。

实时文档翻译支持从英语翻译成任何支持的语言，以及从任何支持的语言翻译成英语。您可以指定源语言代码或使用自动检测。

使用命令行翻译文档

以下示例说明如何通过命令行使用该TranslateDocument操作。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 (^)。

在命令行输入以下命令。

```
aws translate translate-document \
```

```
            --region region \
            --source-language-code "en" \
            --target-language-code "es" \
            --document-content fileb://source-lang.txt
            --document ContentType=text/plain
            --query "TranslatedDocument.Content"
            --output text | base64
            --decode > target-lang.txt
```

命令使用以下 JSON 进行响应：

```
{
    "SourceLanguageCode": "en",
    "TargetLanguageCode": "es",
    "TranslatedDocument":{
      "Content": blob
    }
}
```

使用 Amazon Translate 进行异步批处理

要翻译大量文档（大小不超过 5 GB），请使用 Amazon Translate 异步批处理操作StartTextTranslationJob。此操作最适合短文档集合，如社交媒体帖子或用户评论，或任何不需要即时翻译的情况。

要执行异步批量翻译，通常执行以下步骤：

1. 将一组文档存储在 Amazon S3 存储桶内的输入文件夹中。
2. 启动批量翻译作业。
3. 作为请求的一部分，向 Amazon Translate 提供一个对输入 Amazon S3 文件夹及其所有子文件夹具有读取权限的 IAM 角色。该角色还必须对输出 Amazon S3 存储桶具有读写访问权限。
4. 监控批量翻译作业的进度。
5. 从指定的输出存储桶中检索批量翻译作业的结果。

区域可用性

以下 AWS 区域支持 Batch 翻译：

- 美国东部（俄亥俄州）
- 美国东部（弗吉尼亚州北部）
- 美国西部（加利福尼亚北部）
- 美国西部（俄勒冈州）
- 亚太地区（孟买）
- 亚太地区（首尔）
- 亚太地区（新加坡）
- 亚太地区（悉尼）
- 亚太地区（东京）
- 加拿大（中部）
- 欧洲地区（法兰克福）
- 欧洲地区（爱尔兰）
- 欧洲地区（伦敦）
- 欧洲地区（巴黎）
- 欧洲地区（斯德哥尔摩）

主题

- 批量翻译作业的先决条件
- 运行批量翻译作业
- 监控和分析批量翻译作业
- 获取批量翻译结果

批量翻译作业的先决条件

Amazon Translate 要成功执行批量翻译作业，必须满足以下先决条件：

- 包含输入和输出文档的 Amazon S3 存储桶必须与您调用的 API 端点位于同一 AWS 区域中。
- 批处理输入文档的集合大小必须不超过 5 GB。
- 一个批量翻译作业中最多可提交 100 万份文档。
- 每个输入文档必须不超过 20 MB，并且必须包含少于 100 万个字符。

- 您的文件必须位于 Amazon S3 存储桶中的文件夹中。如果您将输入文件添加到存储桶的顶层，则在您尝试运行批量翻译任务时，Amazon Translate 会引发错误。此要求适用于输入文件。输出文件不需要文件夹，Amazon Translate 可以将其放置于 Amazon S3 存储桶的顶层。
- 您的输入文件文件夹可以包含嵌套文件夹。确保没有任何嵌套文件夹命名为 details，否则在您尝试运行批量翻译任务时，Amazon Translate 会引发错误。

支持的文件格式

Amazon Translate 支持以下类型的文件用于批量翻译作业：

- 纯文本。
- HTML。
- Word 文档 (.docx)。
- PowerPoint 演示文件 (.pptx)。
- Excel 工作簿文件 (.xlsx)。
- XML 本地化交换文件格式 (XLIFF) 文件 (.xlf)。Amazon Translate 仅支持 XLIFF 版本 1.2。

Amazon Translate 要求文件采用 UTF-8 编码。

必备权限

在运行批量翻译任务之前，您的 AWS 账户必须在 IAM 中拥有服务角色。该角色必须具有向 Amazon Translate 授权的权限策略：

- 对 Amazon S3 中您的输入文件夹及其所有子文件夹的读取权限。
- 对输出存储桶的读取和写入访问权限。

还必须包含一项信任策略，允许 Amazon Translate 代入该角色并获取其权限。此信任策略必须允许 `translate.amazonaws.com` 服务主体执行 `sts:AssumeRole` 操作。

在使用 Amazon Translate 控制台创建批量翻译作业时，您可以选择允许 Amazon Translate 自动为您创建此角色。当您使用 AWS CLI 或 Amazon Translate API 运行批量翻译任务时，您需要在请求中提供该角色的亚马逊资源名称 (ARN)。

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。

Example 权限策略

以下示例权限策略授予对 Amazon S3 存储桶中输入文件夹的读取访问权限，并且授予对输出存储桶的读取和写入访问权限。

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

Example 信任策略

以下信任策略允许 Amazon Translate 代入该策略所属的 IAM 角色。

我们建议您验证使用信任策略的 AWS 账户，以缓解 Confused 副手问题。此示例使用 aws:SourceArn 和 aws:SourceAccount 条件键来验证源账户。输入提交批量翻译作业的 AWS 账户。

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "translate.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
            "aws:SourceArn": "arn:aws:translate:*:111122223333:*"
        },
        "StringEquals": {
            "aws:SourceAccount": "111122223333"
        }
      }
    }
  ]
}
```

自定义加密的必备权限

您可以在 Amazon Translate 中自定义您的加密设置，但必须首先在 IAM 中为服务角色添加权限。

Amazon Translate 会对您在运行批量翻译作业时产生的翻译输出进行加密。默认情况下，它使用 AWS 托管式密钥 进行此加密。此类密钥由您的账户创建 AWS 并存储在 AWS Key Management Service (AWS KMS) 中。但是，您无法自己管理此 KMS 密钥，只能由 AWS代表您管理和使用。

或者，您可以选择使用客户托管密钥加密输出，该密钥是您在 AWS 账户中创建、拥有和管理的 KMS 密钥。

您的密钥必须具有允许 Amazon Translate 使用的密钥政策。密钥政策通过将其权限授予允许 Amazon Translate 访问 Amazon S3 存储桶的服务角色来实现这一点。

密钥策略允许服务角色执行加密输出所需的 AWS KMS 操作，如以下示例策略声明所示。

Example KMS 密钥政策语句：

```
{
  "Effect": "Allow",
  "Principal":
  {
```

```
    "AWS":
    [
      "arn:aws:iam::111122223333:role/AmazonTranslateServiceRoleS3FullAccess"
    ]
  },
  "Action":
  [
    "kms:Decrypt",
    "kms:GenerateDataKey",
    "kms:CreateGrant",
    "kms:RetireGrant",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

有关更多信息，请参阅《AWS Key Management Service 开发人员指南》中的 AWS KMS中的密钥政策。

使用其他 AWS 账户 AWS KMS 密钥的权限

如果您想使用与您使用 Amazon Translate 不同的 AWS 账户的 KMS 密钥，则必须：

1. 在 IAM 中更新 Amazon Translate 的服务角色。
2. 更新中的密钥政策 AWS KMS。

要更新您的服务角色，请附加一个允许其使用其他 AWS 账户中的 KMS 密钥执行必要 AWS KMS 操作的策略，如以下示例所示。

Example 用于授予不同账户中 KMS 密钥访问权限的 IAM policy

```
{
  "Effect": "Allow",
  "Action":
  [
    "kms:Decrypt",
    "kms:GenerateDataKey",
    "kms:CreateGrant",
    "kms:RetireGrant",
    "kms:DescribeKey"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:key/key-id"
```

```
}
```

要更新您的 KMS 密钥政策，请将服务角色和管理员用户添加为允许使用密钥的委托人，如以下示例策略语句所示。

Example 允许 IAM 角色使用密钥的 KMS 密钥政策语句

```
{
  "Effect": "Allow",
  "Principal":
  {
    "AWS":
    [
      "arn:aws:iam::444455556666:role/AmazonTranslateServiceRoleS3FullAccess",
      "arn:aws:iam::444455556666:admin"
    ]
  },
  "Action":
  [
    "kms:Decrypt",
    "kms:CreateGrant",
    "kms:GenerateDataKey",
    "kms:RetireGrant",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

有关更多信息，请参阅《AWS Key Management Service 开发者指南》中的允许其他账户中的用户使用 AWS KMS 密钥

运行批量翻译作业

您可以使用 Amazon Translate 控制台 AWS CLI、或 Amazon Translate API 运行批量翻译作业。

> **Note**
>
> 批量翻译作业是长时间运行的操作，可能需要大量时间才能完成。例如，对小型数据集进行批量翻译可能需要几分钟时间，而非常大的数据集可能需要 2 天甚至更长时间。完成时间也取决于资源的可用性。

Amazon Translate 控制台

要使用 Amazon Translate 控制台运行翻译作业，请使用批量翻译页面创建作业：

1. 打开 Amazon Translate 控制台。
2. 在左侧导航菜单中，选择批量翻译。
3. 在翻译作业页面上，选择创建作业。控制台显示创建翻译作业页面。
4. 在作业设置下，执行以下操作：
   a. 在名称中，输入批量翻译作业的自定义名称。
   b. 在源语言中，选择源文件的语言。如果您不知道源文件的语言，或者您的输入文档包含不同源语言，请选择 `auto`。Amazon Translate 会自动检测每个文件的源语言。
   c. 在目标语言中，最多可选择 10 种语言。Amazon Translate 会将每个源文件翻译成每种目标语言。
5. 在输入数据下，执行以下操作：
   a. 在输入 S3 位置中，指定在 Amazon S3 中包含翻译源文件的输入文件夹。要通过在 Amazon S3 中导航到该文件夹来提供该文件夹，请选择选择文件夹。
   b. 在文件格式中，选择翻译源文件的格式。
6. 在输入数据下，执行以下操作：
   a. 在输出 S3 位置中，指定 Amazon Translate 在 Amazon S3 中存放翻译输出的输出文件夹。要通过在 Amazon S3 中导航到该文件夹来提供该文件夹，请选择选择文件夹。
   b. 或者，如果您要使用您在 () 中管理的客户管理的密钥来加密输出，请选择自定义加密设置 AWS Key Management Service （高级AWS KMS）。

      默认情况下，Amazon Translate 使用由 AWS代表您创建、管理和使用的 KMS 密钥。如果您想改用您自己的 KMS 密钥加密输出，请选择此选项。

      如果要使用来自当前 AWS 账户的 KMS 密钥，请在选择 AWS Key Management Service 密钥下将其选中。或者，如果您希望使用来自其他 AWS 账户的 KMS 密钥，请输入该密钥的 Amazon 资源名称 (ARN)。

> **Note**
>
> 在使用您自己的 KMS 密钥之前，您必须在 IAM 中为 Amazon Translate 的服务角色添加权限。如果您想使用其他账户的 KMS 密钥，还必须更新中的密钥策略 AWS KMS。有关更多信息，请参阅 自定义加密的必备权限。

7. 在自定义 - 可选下，您可以选择使用以下设置自定义翻译作业的输出：

   脏话

   掩蔽翻译输出中的污言秽语。如果您为作业指定了多种目标语言，所有目标语言都必须支持脏话掩蔽。如果任何目标语言不支持脏话掩蔽，翻译作业将不会在任何目标语言中掩蔽脏话。

   有关更多信息，请参阅 在 Amazon Translate 中掩蔽污言秽语。

   简洁

   Amazon Translate 不支持批量翻译作业的简洁性。

   有关更多信息，请参阅 在 Amazon Translate 中使用简洁性。

   正式性

   对于某些目标语言，您可以将正式性设置为正式或非正式。如果您为作业指定了多种目标语言，对于任何不支持正式性设置的目标语言，Translate 将忽略该设置。

   有关更多信息，请参阅 在 Amazon Translate 中设置正式性。

   自定义术语

   由示例源术语和每个术语的所需翻译组成。如果您为作业指定了多种目标语言，对于每种在术语文件中具有源术语条目的请求的目标语言，Translate 会为这些语言使用指定的术语。

   有关更多信息，请参阅 使用自定义术语自定义您的翻译。

   并行数据

   由展示您希望如何翻译文本句段的示例组成。如果您为作业指定了多种目标语言，则并行数据文件必须包含所有目标语言的翻译。

   向批量翻译作业添加并行数据时，即创建了一个活动自定义翻译作业。

> **Note**
>
> 与其他不使用并行数据的作业相比，活动自定义翻译作业的定价更高。有关更多信息，请参阅 Amazon Translate 定价。

有关更多信息，请参阅 使用并行数据自定义您的翻译（活动自定义翻译）。

8. 在访问权限下，为 Amazon Translate 提供一个 IAM 角色，该角色向你授予在 Amazon S3 中输入和输出文件所需的权限：
   - 如果您的账户中已有此 IAM 角色，请选择使用现有 IAM 角色，然后在 IAM 角色下选择该角色。
   - 如果您的账户中还没有此 IAM 角色，请选择创建 IAM 角色。在 IAM 角色中，选择输入和输出 S3 存储桶。在角色名称中，提供一个自定义名称。当您创建翻译任务时，Amazon Translate 会自动创建角色。IAM 中的角色名称以 AmazonTranslateServiceRole- 为前缀。

   > **Note**
   >
   > 如果您选择使用自己的 KMS 密钥加密翻译输出，则无法选择创建 IAM 角色。在这种情况下，您必须使用先前存在的 IAM 角色，并且您的 KMS 密钥必须具有允许该角色使用密钥的密钥政策。
   > 有关更多信息，请参阅 自定义加密的必备权限

9. 请选择创建任务。

   控制台会返回翻译作业页面，其中的作业创建状态显示在页面顶部的横幅中。几分钟后，表中会显示您的作业。

10. 在名称列中选择作业名称，以打开作业详细信息页面。

    您的翻译作业运行时，状态字段会显示正在进行。

11. 当状态变为已完成时，选择输出文件位置下的链接进入翻译输出。控制台会转到您在 Amazon S3 中的输出存储桶。
12. 要下载输出文件，请选中每个文件的复选框，然后选择下载。

AWS CLI

要使用运行翻译作业 AWS CLI，请使用`start-text-translation-job`命令，然后为`parallel-data-names`参数指定 parallel 数据资源的名称。

Example Start-text-translation-job 命令

以下示例通过提交存储在 Amazon S3 中输入存储桶中的 Excel 文件来运行翻译任务。此任务由请求中包含的并行数据进行自定义。

```
$ aws translate start-text-translation-job \
> --input-data-config ContentType=application/vnd.openxmlformats-officedocument.spreadsheetml.sheet,S3Uri=s3://amzn-s3-demo-bucket/input/ \
> --output-data-config S3Uri=s3://amzn-s3-demo-bucket/output/ \
> --data-access-role-arn arn:aws:iam::111122223333:role/my-iam-role \
> --source-language-code en \
> --target-language-codes es it \
> --job-name my-translation-job
```

如果命令成功，Amazon Translate 将返回作业 ID 和状态：

```
{
    "JobId": "4446f95f20c88a4b347449d3671fbe3d",
    "JobStatus": "SUBMITTED"
}
```

如果您想要自定义翻译作业的输出，可以使用以下参数：

`--settings`

用于配置翻译输出的设置，包括以下选项：

在翻译输出中开启简洁功能。Amazon Translate 不支持批量翻译作业的简洁性。有关更多信息，请参阅 在 Amazon Translate 中使用简洁性。

启用脏话设置以掩蔽污言秽语。要启用该设置，请将脏话参数设置为 `Profanity=MASK`。有关更多信息，请参阅 在 Amazon Translate 中掩蔽污言秽语。如果任何目标语言不支持脏话掩蔽，翻译作业将不会掩蔽任何目标语言中的脏话。

在翻译输出中设置正式程度。将 `Formality` 参数设置为 `FORMAL` 或 `INFORMAL`。如果您为作业指定了多种目标语言，对于任何不支持正式性设置的目标语言，Translate 将忽略该设置。有关更多信息，请参阅 在 Amazon Translate 中设置正式性。

`--terminology-names`

要添加到翻译作业中的自定义术语资源的名称。此资源列出了示例资源术语和每个术语所需的翻译。如果您为作业指定了多种目标语言，对于每种在术语文件中具有源术语条目的请求的目标语言，Translate 会为这些语言使用指定的术语。

此参数仅接受一个自定义术语资源。

要查看可用的自定义术语资源列表，请使用 `list-terminologies` 命令。

有关更多信息，请参阅 使用自定义术语自定义您的翻译。

`--parallel-data-names`

要添加到翻译作业的并行数据资源的名称。此资源由展示您希望如何翻译文本句段的示例组成。如果您为作业指定了多种目标语言，则并行数据文件必须包含所有目标语言的翻译。

向翻译作业添加并行数据时，即创建了一个活动自定义翻译作业。

此参数仅接受一个并行数据资源。

> **Note**
>
> 与其他不使用并行数据的作业相比，活动自定义翻译作业的定价更高。有关更多信息，请参阅 Amazon Translate 定价。

要查看可用并行数据资源的列表，请使用 `list-parallel-data` 命令。

有关更多信息，请参阅 使用并行数据自定义您的翻译（活动自定义翻译）。

要查看翻译作业，请使用 `describe-text-translation-job` 命令。

Example Describe-text-translation-job 命令

以下示例通过提供作业 ID 来检查作业状态。此 ID 由 Amazon Translate 在使用 `start-text-translation-job` 命令启动作业时提供的。

```
$ aws translate describe-text-translation-job \
> --job-id 4446f95f20c88a4b347449d3671fbe3d
```

Amazon Translate 使用任务属性进行响应，其中包括其状态：

```
{
    "TextTranslationJobProperties": {
        "JobId": "4446f95f20c88a4b347449d3671fbe3d",
        "JobName": "my-translation-job",
        "JobStatus": "COMPLETED",
        "JobDetails": {
            "TranslatedDocumentsCount": 0,
            "DocumentsWithErrorsCount": 0,
            "InputDocumentsCount": 1
        },
        "SourceLanguageCode": "en",
        "TargetLanguageCodes": [
            "es",
            "it"
        ],
        "SubmittedTime": 1598661012.468,
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/input/",
            "ContentType": "application/vnd.openxmlformats-
officedocument.spreadsheetml.sheet"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/output/111122223333-
TranslateText-4446f95f20c88a4b347449d3671fbe3d/"
        },
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/my-iam-role"
    }
}
```

如果 `JobStatus` 值为 `IN_PROGRESS`，请等待几分钟，然后再次运行 `describe-text-translation-job`，直到状态变为 `COMPLETED`。作业完成后，您可以在 `OutputDataConfig` 下的 `S3Uri` 字段提供的位置中下载翻译结果。

Amazon Translate API

要使用 Amazon Translate API 提交批量翻译作业，请使用StartTextTranslationJob操作。

监控和分析批量翻译作业

您可以使用作业的 ID 来监控其进度并获取其输出文档的 Amazon S3 位置。要监视特定作业，请使用 DescribeTextTranslationJob 操作。您还可以使用 ListTextTranslationJobs 操作检索账户中所有翻译

作业的信息。要将结果限制为符合特定条件的作业，请使用ListTextTranslationJobs操作的filter参数。您可以按作业名称、作业状态或提交作业的日期和时间筛选结果。

Example describe-text-translation-job 命令

以下示例使用运行DescribeTextTranslationJob命令来检查作业的状态： AWS CLI

```
$ aws translate describe-text-translation-job --job-id 1c1838f470806ab9c3e0057f14717bed
```

此命令将返回以下输出：

```
{
  "TextTranslationJobProperties": {
    "InputDataConfig": {
      "ContentType": "text/plain",
      "S3Uri": "s3://input-bucket-name/folder"
    },
    "EndTime": 1576551359.483,
    "SourceLanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::012345678901:role/service-role/
AmazonTranslateInputOutputAccess",
    "JobId": "1c1838f470806ab9c3e0057f14717bed",
    "TargetLanguageCodes": [
      "fr"
    ],
    "JobName": "batch-test",
    "SubmittedTime": 1576544017.357,
    "JobStatus": "COMPLETED",
    "Message": "Your job has completed successfully.",
    "JobDetails": {
      "InputDocumentsCount": 77,
      "DocumentsWithErrorsCount": 0,
      "TranslatedDocumentsCount": 77
    },
    "OutputDataConfig": {
      "S3Uri": "s3://bucket-name/output/012345678901-
TranslateText-1c1838f470806ab9c3e0057f14717bed/"
    }
  }
}
```

您可以通过使用 StopTextTranslationJob 操作在批处理翻译作业的状态为 IN_PROGRESS 时停止此作业。

Example stop-text-translation-job 命令

以下示例使用运行StopTextTranslationJob命令来停止批量翻译： AWS CLI

```
$ aws translate stop-text-translation-job --job-id 5236d36ce5192abdb3e2519f3ab8b065
```

此命令将返回以下输出：

```
{
  "TextTranslationJobProperties": {
    "InputDataConfig": {
      "ContentType": "text/plain",
      "S3Uri": "s3://input-bucket-name/folder"
    },
    "SourceLanguageCode": "en",
    "DataAccessRoleArn": "arn:aws:iam::012345678901:role/service-role/
AmazonTranslateInputOutputAccess",
    "TargetLanguageCodes": [
      "fr"
    ],
    "JobName": "canceled-test",
    "SubmittedTime": 1576558958.167,
    "JobStatus": "STOP_REQUESTED",
    "JobId": "5236d36ce5192abdb3e2519f3ab8b065",
    "OutputDataConfig": {
      "S3Uri": "s3://output-bucket-name/012345678901-
TranslateText-5236d36ce5192abdb3e2519f3ab8b065/"
    }
  }
}
```

获取批量翻译结果

作业的状态为 `COMPLETED` 或 `COMPLETED_WITH_ERROR` 后，您的输出文档将在您指定的 Amazon S3 文件夹中可用。输出文档名称与输入文档名称相匹配，但添加目标语言代码作为前缀。例如，如果您将一个名为 `mySourceText.txt` 的文档翻译成法语，则输出文档将名为 `fr.mySourceText.txt`。

如果批处理翻译作业的状态为 `FAILED`，则 DescribeTextTranslationJob 操作响应包括一个 `Message` 字段，用于描述作业未成功完成的原因。

每个批量翻译作业还会生成一个辅助文件，其中包含有关所执行翻译的信息，例如翻译的总字符数和遇到的错误数。此文件（称为 *target-language-code*.auxiliary-translation-details.json）在输出文件夹的 details 子文件夹中生成。

以下是批量翻译辅助文件的示例。

```
{
  "sourceLanguageCode": "en",
  "targetLanguageCode": "fr",
  "charactersTranslated": "105",
  "documentCountWithCustomerError": "0",
  "documentCountWithServerError": "0",
  "inputDataPrefix": "s3://input-bucket-name/folder",
  "outputDataPrefix": "s3://output-bucket-name/012345678901-
TranslateText-1c1838f470806ab9c3e0057f14717bed/",
  "details": [
    {
      "sourceFile": "mySourceText.txt",
      "targetFile": "fr.mySourceText.txt",
      "auxiliaryData": {
        "appliedTerminologies": [
          {
            "name": "TestTerminology",
            "terms": [
              {
                "sourceText": "Amazon",
                "targetText": "Amazon"
              }
            ]
          }
        ]
      }
    },
    {
      "sourceFile": "batchText.txt",
      "targetFile": "fr.batchText.txt",
      "auxiliaryData": {
        "appliedTerminologies": [
          {
            "name": "TestTerminology",
            "terms": [
              {
                "sourceText": "Amazon",
```

```
                "targetText": "Amazon"
              }
            ]
          }
        ]
      }
    }
  ]
}
```

使用 Amazon Translate 自定义您的翻译

您可以在 Amazon Translate 中使用以下设置自定义您生成的翻译：

- Do-not-translate 标签-使用开始和结束标签来指定您不想翻译的内容（在 HTML 内容中）。
- 自定义术语 — 定义您希望 Amazon Translate 如何翻译特定术语，例如品牌名称。
- 简洁 — 缩短大多数翻译的翻译输出的长度（与不简洁的翻译输出相比）。在实时文本翻译中简洁明了。
- 亵渎 — 在翻译输出中掩盖亵渎词语和短语。
- 形式化 — 在翻译输出中设置语言形式化级别。
- 并行数据-调整翻译输出，以反映您提供的示例翻译样本中的风格、语气和单词选择。

主题

- do-not-translate在亚马逊 Translate 中使用
- 使用自定义术语自定义您的翻译
- 在 Amazon Translate 中使用简洁性
- 在 Amazon Translate 中掩蔽污言秽语
- 在 Amazon Translate 中设置正式性
- 使用并行数据自定义您的翻译（活动自定义翻译）

do-not-translate在亚马逊 Translate 中使用

对于 HTML 内容，您可以为不想翻译的文本添加 do-not-translate标签。此功能可用于控制台和 API 操作。

主题

- 在控制 do-not-translate台上使用
- do-not-translate与 API 一起使用

在控制 do-not-translate台上使用

在源 HTML 内容中，`translate="no"`在您不想翻译的内容周围的 HTML 标签中指定。例如，将以下文本从英语翻译成德语：

```
In French, the Louvre Museum is Musée du Louvre.
```

文本 “卢浮宫博物馆” 需要保留法语，因此我们使用 span 标签来跳过此内容的翻译：

```
<p>In French, the Louvre Museum  is <span translate="no">Musée du Louvre</span>.</p>
```

这句话的结果是德语翻译：

```
<p>Auf Französisch ist <span translate="no">Musée du Louvre</span> das Louvre-Museum.</
p>
```

do-not-translate与 API 一起使用

您可以 do-not-translate与实时 API 操作（`TranslateText`和`TranslateDocument`）和异步 `StartTextTranslationJob` API 操作一起使用。在您为 API 请求提供的源文本中，您可以使用任何类型的 HTML 元素来指定需要跳过翻译的内容。

在以下示例中，我们想将一些文本从英语翻译成西班牙语，但保留一些英文文本：

```
aws translate translate-text \
  --source-language-code "en" \
  --target-language-code "es" \
  --region us-west-2 \
  --text "<p>You can translate this paragraph to any language.</p> <p translate=no>But
 do not translate this.</p>"
```

此 API 请求将返回以下西班牙语翻译：

```
{
    "TranslatedText": "<p>Puede traducir este párrafo a cualquier idioma.</p>
                            <p translate=no>But do not translate this.</p>",
    "SourceLanguageCode": "en",
    "TargetLanguageCode": "es"
}
```

使用自定义术语自定义您的翻译

在翻译请求中使用自定义术语，确保您的品牌名称、角色名称、型号名称和其他独特内容翻译成所需的结果。

您可以创建术语文件并将其上传到您的 Amazon Translate 账户。有关文件大小和术语文件数量的信息，请参阅服务限额。翻译文本时，您可以选择要使用的自定义术语文件。当 Amazon Translate 发现源文本和术语文件之间存在匹配项时，会使用术语文件中的翻译。

考虑以下示例：Amazon Photos provides free photo and video storage to Amazon Prime members. 在法语中，该名称不翻译：仍为 Amazon Photos。

当您在没有任何其他上下文的情况下使用 Amazon Translate 将 Amazon Photos 翻译成法语时，结果是 Photos d'Amazon，这不是您所需的翻译。

如果您为 Amazon Photos 一词添加自定义术语条目，指定法语翻译为 Amazon Photos，则 Amazon Translate 会使用自定义术语将该短语翻译成所需的结果。

Amazon Translate 不保证每次翻译都会使用目标术语。自定义术语使用翻译上下文中源术语和目标术语的含义来决定是否使用目标术语。有关更多信息，请参阅 最佳实践。

主题

- 创建自定义术语
- 使用自定义术语
- 使用自定义术语示例 AWS SDK for Python (Boto)
- 加密术语
- 最佳实践

创建自定义术语

您可以通过创建术语文件来定义自定义术语。Amazon Translate 支持 CSV、TSV 或 TMX 文件格式。文件中的每个条目都包含每种目标语言的源术语和等效（已翻译）术语。

在创建术语文件后，您可以将文件上传到您的 Amazon Translate 账户。

> **Important**
>
> 自定义术语中的源文本区分大小写。在翻译过程中，Amazon Translate 会在输入文档中找到完全匹配的术语时使用自定义术语。

术语文件格式

以下示例显示 CSV 格式的术语文件。

CSV（逗号分隔值）

```
en,fr,es
Amazon Photos,Amazon Photos,Amazon Photos
```

以下示例显示 TMX 格式的术语文件。TMX 文件使用翻译软件经常使用的 XML 格式。

TMX（翻译记忆库交换）

```
<?xml version="1.0" encoding="UTF-8"?>
 <tmx version="1.4">
  <header
     creationtool="XYZTool" creationtoolversion="0"
     datatype="PlainText" segtype="sentence"
     adminlang="en-us" srclang="en"
     o-tmf="test"/>
  <body>
    <tu>
      <tuv xml:lang="en">
        <seg>Amazon Photos</seg>
      </tuv>
      <tuv xml:lang="fr">
        <seg>Amazon Photos</seg>
      </tuv>
      <tuv xml:lang="es">
        <seg>Amazon Photos</seg>
      </tuv>
    </tu>
  </body>
 </tmx>
```

方向性

上传自定义术语文件时，需要为自定义术语设置方向性 值。方向性表示术语文件指定一种源语言还是多种源语言。

对于方向性，设置为下列值之一：

单向

术语文件包含一种源语言（列表中的第一种语言）。所有其他语言均为目标语言。

例如，在 CSV 文件中，第一列包含源语言的文本，所有其他列包含目标语言的文本。

多向

文件中的任何语言都可以是源语言或目标语言。例如，如果您的术语文件包含英语、法语和西班牙语文本，则可以将该文件用于翻译以下语言对的作业：

- 英语到法语
- 英语到西班牙语
- 法语到英语
- 法语到西班牙语
- 西班牙语到英语
- 西班牙语到法语

相比之下，您需要为这六个翻译作业创建三个单向术语文件（每种源语言一个）。

使用自定义术语

要在使用TranslateText操作翻译文本时使用自定义术语，请包括可选TerminologyNames参数。

例如，如果您将以下名为 Amazon_Family.csv 的术语文件上传到您的账户：

```
en,fr
Amazon Family,Amazon Famille
```

您可以通过自定义术语使用以下 CLI 命令翻译文本。

> **Note**
>
> 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 translate translate-text \
     --region region \
     --source-language-code "en" \
     --target-language-code "fr" \
     --terminology-names "Amazon_Family" \
     --text "Have you ever stored videos in Amazon Family?"
```

此操作会使用选定的自定义术语将文本翻译为“Avez-vous déjà fait des achats avec Amazon Famille?”而非直接翻译为“Avez-vous déjà fait des achats avec Famille Amazon?”，显然后者并不是我们想要的翻译。

以下示例说明了如何在 Python 中使用相同的术语文件。

```
import boto3

translate = boto3.client(service_name='translate')

print("Translating 'Have you ever shopped with Amazon Family?' from English to French
 with the 'Amazon_Family' custom terminology...")
response = translate.translate_text(Text="Have you ever shopped with Amazon Family?",
 TerminologyNames=["Amazon_Family"], SourceLanguageCode="en", TargetLanguageCode="fr")
print("Translated text: " + response.get('TranslatedText'))
print("\n")
```

有关使用自定义术语进行 Amazon Translate 操作的更多信息，请参阅 API 操作。

使用自定义术语示例 AWS SDK for Python (Boto)

以下示例说明如何在 Python 中使用自定义术语操作。要运行此示例，请通过安装 Python 软件开发工具包 AWS CLI。有关说明，请参阅 the section called “设置 AWS CLI”。

```
    #!/usr/bin/env python
    # -*- coding: utf-8 -*-

    import boto3

    translate = boto3.client(service_name='translate')

    # The terminology file 'my-first-terminology.csv' has the following contents:
    '''
    en,fr
    Amazon Family,Amazon Famille
    '''

    # Read the terminology from a local file
    with open('/tmp/my-first-terminology.csv', 'rb') as f:
        data = f.read()

    file_data = bytearray(data)
```

```
    print("Importing the terminology into Amazon Translate...")
    response = translate.import_terminology(Name='my-first-terminology',
 MergeStrategy='OVERWRITE', TerminologyData={"File": file_data, "Format": 'CSV'})
    print("Terminology imported: "),
    print(response.get('TerminologyProperties'))
    print("\n")

    print("Getting the imported terminology...")
    response = translate.get_terminology(Name='my-first-terminology',
 TerminologyDataFormat='CSV')
    print("Received terminology: "),
    print(response.get('TerminologyProperties'))
    print("The terminology data file can be downloaded here: " +
 response.get('TerminologyDataLocation').get('Location'))
    print("\n")

    print("Listing the first 10 terminologies for the account...")
    response = translate.list_terminologies(MaxResults=10)
    print("Received terminologies: "),
    print(response.get('TerminologyPropertiesList'))
    print("\n")

    print("Translating 'Amazon Family' from English to French with no terminology...")
    response = translate.translate_text(Text="Amazon Family", SourceLanguageCode="en",
 TargetLanguageCode="fr")
    print("Translated text: " + response.get('TranslatedText'))
    print("\n")

    print("Translating 'Amazon Family' from English to French with the 'my-first-
terminology' terminology...")
    response = translate.translate_text(Text="Amazon Family", TerminologyNames=["my-
first-terminology"], SourceLanguageCode="en", TargetLanguageCode="fr")
    print("Translated text: " + response.get('TranslatedText'))
    print("\n")

    # The terminology file 'my-updated-terminology.csv' has the following contents:
    '''
    en,fr
    Amazon Family,Amazon Famille
    Prime Video, Prime Video
    '''

    # Read the terminology from a local file
    with open('/tmp/my-updated-terminology.csv', 'rb') as f:
```

```
        data = f.read()

    file_data = bytearray(data)

    print("Updating the imported terminology in Amazon Translate...")
    response = translate.import_terminology(Name='my-first-terminology',
 MergeStrategy='OVERWRITE', TerminologyData={"File": file_data, "Format": 'CSV'})
    print("Terminology updated: "),
    print(response.get('TerminologyProperties'))
    print("\n")

    print("Translating 'Prime Video' from English to French with no terminology...")
    response = translate.translate_text(Text="Prime Video", SourceLanguageCode="en",
 TargetLanguageCode="fr")
    print("Translated text: " + response.get('TranslatedText'))
    print("\n")

    print("Translating 'Prime Video' from English to French with the 'my-first-
terminology' terminology...")
    response = translate.translate_text(Text="Prime Video", TerminologyNames=["my-
first-terminology"], SourceLanguageCode="en", TargetLanguageCode="fr")
    print("Translated text: " + response.get('TranslatedText'))
    print("\n")

    print("Cleaning up by deleting 'my-first-terminology'...")
    translate.delete_terminology(Name="my-first-terminology")
    print("Terminology deleted.")
```

加密术语

Amazon Translate 致力于保护您的所有数据，您的自定义术语也不例外。每个自定义术语在创建时都会加密，因此只能由您访问。

有三个可用的加密选项：

- 使用 AWS 加密。 AWS 加密是保护您的信息的默认选项。
- 使用与您的账户关联的加密密钥。控制台中的一个菜单将为您提供可供使用的关联加密密钥选项。
- 使用与您的账户不关联的加密密钥。控制台会显示一个输入字段，供您输入加密密钥的 Amazon 资源名称 (ARN)。

最佳实践

使用自定义术语时，请使用以下一般最佳做法：

- 保持您的自定义术语整洁。仅包括需要控制其翻译值的术语。
- 自定义术语区分大小写。如果您要为单词的大写和非大写版本提供目标翻译，请为每个版本各添加一个条目。
- 自定义术语并不能作为用来控制间距、标点符号或大小写的工具。例如，避免使用以下类型的条目：
  - 添加空格 — 英语：美国 FR：U S A
  - 添加标点符号 — 英语：美国 FR：U.S.A
  - 更改大写 — 英语：美国 FR：美国
- 不要为同一个源短语添加多个不同的翻译。例如：
  - 条目 #1 – EN: Amazon FR: Amazon
  - 条目 #2 – EN: Amazon FR: Amazone
- 您可以为 Amazon Translate 支持的任何语言创建自定义术语文件。

Amazon Translate 不保证自定义术语在每次翻译中都使用目标术语。要提高自定义术语的准确性，请在为术语文件创建内容时遵循以下最佳实践：

- 自定义术语适用于任何词汇，包括动词和同形异义词。专有名词（例如品牌名称和产品名称）是理想条目。
- 目标术语应使用流畅的目标语言。对于包含大量特殊字符或格式的目标术语，建议不要使用自定义术语。
- 您可以在术语文件中包含多字短语或子句。但是，包含多个单词的术语不太可能在目标语言中实现流畅阅读。
- 自定义术语使用翻译上下文中源术语和目标术语的含义来决定是否使用目标术语。如果目标术语不适合给定的翻译环境，Amazon Translate 可能不会使用该目标术语。

  例如，如果您的术语文件包含以下英语到法语的条目：

  `EN: order, FR: commande`(根据英文 “to order” 翻译成法语 “指挥官”).

  Amazon Translate 在翻译以下句子时不使用此条目，因为翻译上下文不匹配：

  “为了让我们帮助你，请分享你的名字。”

避免这种情况的建议：

- 确保每种语言的目标术语在语义上与源术语相同。
- 避免使用具有多种含义的源词或目标术语。

在 Amazon Translate 中使用简洁性

在不同语言之间进行翻译时，有时翻译输出（按字符数计算）会比预期的长。如果没有空间容纳额外字符，则在某些情况下（例如字幕、字幕、标题或表单字段），较长的输出可能会导致问题。

当你使用 Amazon Translate 运行实时文本翻译时，你可以开启简洁设置。简洁可缩短大多数翻译的翻译输出的长度（与不简洁的翻译输出相比）。

主题

- 使用简洁设置
- 支持的语言

使用简洁设置

您可以将简洁设置与实时文本翻译配合使用。Amazon Translate 不支持实时文档翻译或异步翻译作业的简洁性。

要在实时文本翻译请求中简洁起见，请执行以下任一操作：

- 控制台-在 “实时翻译” 页面的 “文本” 选项卡的 “其他设置” 下，选择 “简洁” 设置。
- AWS CLI— 在`translate-text`命令中，将`--settings`参数的简洁性设置为。`Brevity=ON`有关更多信息，请参阅《AWS CLI 命令参考》中的 translate-text。
- AWS API — 在 TranslateTextAPI 操作中，在 “设置” 参数中配置简洁性。

支持的语言

Amazon Translate 支持以下语言对的简洁性：

- 从任何源语言到下表中的一种语言。
- 从下表中的任何一种语言到英语。

语言	语言代码
法语	fr
德语	de
意大利语	it
巴西葡萄牙语	pt
西班牙语	es

如果您要求使用不支持的语言对进行简洁翻译，则在关闭简洁设置的情况下继续翻译。

在 Amazon Translate 中掩蔽污言秽语

在使用 Amazon Translate 运行翻译时，您可以启用脏话设置来掩蔽翻译输出中的污言秽语。

为了掩蔽污言秽语，Amazon Translate 会将其替换为 Grawlix 字符串“?$#@$”。这 5 字符组成的序列会用于每个污言秽语，无论词汇的长度或数量如何。

在源语言和目标语言相同的翻译请求中，Amazon Translate 不会掩蔽脏话。

在某些情况下，源输入中的污言秽语在翻译后的输出中会自然而然地变得没有冒犯之意。在这种情况下，不会应用掩蔽。

Amazon Translate 按字面意思检测每个污言秽语，而非根据上下文进行检测。这意味着如果某个词汇被认为是污言秽语，那么即使该词汇在上下文中并没有冒犯之意，也可能遭到掩蔽。例如，如果 Amazon Translate 检测到“jerk”是污言秽语，那么它会把“jerk chicken”这一短语写作“?$#@$ chicken”，尽管“jerk chicken”其实并无冒犯之意。（此处“jerk”仅用作示例。Amazon Translate 不会将该词汇检测为脏话。）

主题

- 使用脏话设置
- 语言不受支持

使用脏话设置

在 Amazon Translate 中，您可以将脏话设置与两种类型的翻译操作结合使用：实时翻译和异步批处理。

要掩蔽实时翻译请求中的脏话，请执行以下任一操作：

- 在 Amazon Translate 控制台的实时翻译页面上，在其他设置中，启用脏话设置。
- 在的`translate-text`命令中 AWS CLI，将`--settings`参数设置为`Profanity=MASK`。有关更多信息，请参阅《AWS CLI 命令参考》中的 translate-text。
- 在 Amazon Translate API 的`TranslateText`操作中，使用`Settings`参数设置亵渎掩盖。有关更多信息，请参阅 API 参考TranslateText中的。

要掩蔽异步批处理操作中的脏话，请参阅运行批量翻译作业。

语言不受支持

在翻译成 Amazon Translate 支持的任何目标语言时，您可以掩蔽脏话，以下语言除外：

语言	语言代码
孟加拉语	bn
印地语	hi
马来亚拉姆语	ml
旁遮普语	pa
僧伽罗语	si
越南语	vi

有关 Amazon Translate 支持的所有语言，请参阅支持的语言和语言代码。

在 Amazon Translate 中设置正式性

您可以选择为支持的目标语言的翻译指定所需的正式性 程度。正式性设置控制翻译输出中所使用语言的正式程度（也称为 敬语 或 语域）。正式性设置可用于实时翻译和异步批处理。

正式性支持以下值：

- 非正式 – 翻译文本中的所有句子都使用与非正式交流相关的语言结构。例如，翻译后的文本使用非正式的第二人称代词形式以及相应的动词形式（或日语的 Kudaketa 形式）。
- 正式 – 翻译文本中的所有句子都使用与正式、礼貌交流相关的语言结构。例如，翻译后的文本使用正式的第二人称代词形式以及相应的动词形式（或日语的 Teineigo 形式）。

例如，“Are you sure?”这句话 可以有两个正确的德语翻译：用于正式语域的“Sind Sie sicher?” 和用于非正式语域的 “Bist du sicher?”。

如果 Amazon Translate 不支持目标语言的正式程度，或者您未指定正式性参数，则翻译作业将忽略正式性设置。

主题

- 使用正式性设置
- 支持的语言

使用正式性设置

若要在实时翻译请求中设置正式性，请执行以下操作之一：

- 在 Amazon Translate 控制台的实时翻译页面上，在其他设置中，启用正式性设置并选择一个值。
- 在 Amazon Translate API 的TranslateText操作中使用设置参数。
- 对于中的`translate-text`命令 AWS CLI，将`--settings`参数设置为`Formality=FORMAL`或`Formality=INFORMAL`。有关更多信息，请参阅《AWS CLI 命令参考》中的 translate-text。

要在批量翻译请求中设置正式性，请在开始翻译作业时设置正式性参数。有关详细信息和示例，请参阅运行批量翻译作业。

对于 CLI 或 API 请求，响应中的 `AppliedSettings` 字段包含请求中的正式性设置（如有）。如果目标语言不支持正式性，则响应中的 `AppliedSettings` 值为 NULL。

支持的语言

Amazon Translate 支持从任何源语言翻译成以下目标语言的正式性设置。

语言	语言代码
荷兰语	nl
法语	fr
法语（加拿大）	fr-CA
德语	de
印地语	hi
意大利语	it
日语	ja
韩语	ko
葡萄牙语（葡萄牙）	pt-PT
西班牙语	es
西班牙语（墨西哥）	es-MX

有关 Amazon Translate 支持的所有语言，请参阅 支持的语言和语言代码。

使用并行数据自定义您的翻译（活动自定义翻译）

向 Amazon Translate 添加并行数据，以自定义批量翻译任务的输出。并行数据由展示您希望如何翻译文本句段的示例组成。其中包含源语言的文本示例集合，对于每个示例，都包含一种或多种目标语言的期望翻译输出。

向批量翻译作业添加并行数据时，即创建了一个活动自定义翻译作业。在运行这些作业时，Amazon Translate 会在运行时系统中使用您的并行数据来生成自定义机器翻译输出。该服务可调整其翻译，以反映在您的并行数据中发现的风格、语气和词汇选择。借助并行数据，您可以为生命科学、法律或金融等特定领域特有的术语或短语定制翻译。

> **Note**
>
> 与其他不使用并行数据的作业相比，活动自定义翻译作业的定价更高。有关更多信息，请参阅 Amazon Translate 定价。

例如，CSV 文件中定义了以下并行数据：

```
"en","fr"
"How are you?","Comment ça va ?"
```

在此示例中，英语 (en) 是源语言，法语 (fr) 是目标语言。该示例显示了源短语“How are you?” 应如何翻译成法语。将此示例输入文件导入 Amazon Translate 后，可以将其应用于翻译作业以影响其输出。在这类作业中，Amazon Translate 会将“How are you?” 翻译为非正式的“Comment ça va?”，而不是正式的“Comment allez-vous?” 例如，该作业可能会收到以下源文本：

```
Hello, how are you?
How are you?
Hi, how are you?
How are you doing?
```

从这段文本，该作业生成以下翻译：

```
Bonjour, comment ça va ?
Comment ça va ?
Salut, comment ça va ?
Comment ça va ?
```

相比之下，如果作业在没有并行数据的情况下运行，则输出中可能会包含更为正式的“comment allez-vous”：

```
Bonjour, comment allez-vous ?
Comment allez-vous ?
Salut, comment allez-vous ?
Comment allez-vous ?
```

通过借助并行数据自定义批量翻译作业，就像是使用您自己的翻译示例训练的自定义翻译模型进行翻译，可以影响输出。使用活动自定义翻译，您无需再训练自定义模型，而且可以避免此类训练所需的时

间和费用。如果您的翻译要求随着时间的推移而发生变化，您可以通过更新并行数据来优化输出，这比重新训练自定义模型要容易得多。

区域可用性

活动自定义翻译在以下区域可用：

- 美国东部（弗吉尼亚州北部）
- 美国西部（俄勒冈州）
- 欧洲地区（爱尔兰）

主题

- Amazon Translate 的并行数据输入文件
- 将您的并行数据添加到 Amazon Translate
- 在 Amazon Translate 中查看和管理您的并行数据

Amazon Translate 的并行数据输入文件

在 Amazon Translate 中创建并行数据资源之前，您必须先创建一个包含翻译示例的输入文件。您的并行数据输入文件必须使用 Amazon Translate 支持的语言。有支持语言的列表，请参阅 支持的语言和语言代码。

并行数据示例

下表中的文本展示了可以将格式转换为并行数据输入文件的翻译句段的示例：

en	es
Amazon Translate is a neural machine translation service.	Amazon Translate es un servicio de traducción automática basado en redes neuronales.
Neural machine translation is a form of language translation automation that uses deep learning models.	La traducción automática neuronal es una forma de automatizar la traducción de lenguajes utilizando modelos de aprendizaje profundo.

en	es
Amazon Translate allows you to localize content for international users.	Amazon Translate le permite localizar contenido para usuarios internacionales.

表的第一行提供语言代码。第一个语言英语 (en) 是源语言。目标语言为西班牙语。第一列提供了源文本的示例。其他列包含翻译示例。当使用这些并行数据自定义批处理作业时，Amazon Translate 会调整翻译以反映示例。

输入文件格式

Amazon Translate 支持以下格式的并行数据输入文件：

- 翻译记忆库交换 (TMX)
- 逗号分隔值 (CSV)
- 制表符分隔值 (TSV)

TMX

Example TMX 输入文件

以下示例 TMX 文件以 Amazon Translate 可接受的格式定义并行数据。在此文件中，英语 (en) 是源语言。目标语言为西班牙语 (es)。作为并行数据的输入文件，该文件提供了几个可供 Amazon Translate 用于定制批处理任务输出的示例。

```
<?xml version="1.0" encoding="UTF-8"?>
<tmx version="1.4">
    <header srclang="en"/>
    <body>
        <tu>
            <tuv xml:lang="en">
                <seg>Amazon Translate is a neural machine translation service.</seg>
            </tuv>
            <tuv xml:lang="es">
                <seg>Amazon Translate es un servicio de traducción automática basado
 en redes neuronales.</seg>
            </tuv>
        </tu>
```

```
        <tu>
            <tuv xml:lang="en">
                <seg>Neural machine translation is a form of language translation
 automation that uses deep learning models.</seg>
            </tuv>
            <tuv xml:lang="es">
                <seg>La traducción automática neuronal es una forma de automatizar
 la traducción de lenguajes utilizando modelos de aprendizaje profundo.</seg>
            </tuv>
        </tu>
        <tu>
            <tuv xml:lang="en">
                <seg>Amazon Translate allows you to localize content for
 international users.</seg>
            </tuv>
            <tuv xml:lang="es">
                <seg>Amazon Translate le permite localizar contenido para usuarios
 internacionales.</seg>
            </tuv>
        </tu>
    </body>
</tmx>
```

TMX 要求

在 TMX 文件中定义并行数据时，请记住 Amazon Translate 的以下要求：

- Amazon Translate 支持 TMX 1.4b。有关更多信息，请参阅全球化和本地化协会网站上的 TMX 1.4b 规范。
- `header` 元素必须包含 `srclang` 属性。此属性的值决定并行数据的源语言。
- `body` 元素必须包含至少一个翻译单元 (`tu`) 元素。
- 每个 `tu` 元素必须包含至少两个翻译单元变体 (`tuv`) 元素。其中一个 `tuv` 元素的 `xml:lang` 属性值必须与 `header` 元素中分配给 `srclang` 属性的值相同。
- 所有 `tuv` 元素都必须具有 `xml:lang` 属性。
- 所有 `tuv` 元素都必须有句段 (`seg`) 元素。
- 在处理您的输入文件时，如果遇到空白或仅包含空格的 `seg` 元素，Amazon Translate 会跳过某些 `tu` 或 `tuv` 元素：
  - 如果 `seg` 元素对应于源语言，则 Amazon Translate 会跳过 `seg` 元素占用的 `tu` 元素。
  - 如果 `seg` 元素对应于目标语言，则 Amazon Translate 会跳过 `seg` 元素占用的 `tuv` 元素。

- 在处理您的输入文件时，如果遇到超过 1000 字节的 `seg` 元素，Amazon Translate 会跳过某些 `tu` 或 `tuv` 元素：
  - 如果 `seg` 元素对应于源语言，则 Amazon Translate 会跳过 `seg` 元素占用的 `tu` 元素。
  - 如果 `seg` 元素对应于目标语言，则 Amazon Translate 会跳过 `seg` 元素占用的 `tuv` 元素。
- 如果输入文件包含多个具有相同源文本的 `tu` 元素，Amazon Translate 会执行以下操作之一：
  - 如果 `tu` 元素具有 `changedate` 属性，它会使用具有最新日期的元素。
  - 否则，它会使用最接近文件末尾的元素。

CSV

以下示例 CSV 文件以 Amazon Translate 可接受的格式定义并行数据。在此文件中，英语 (en) 是源语言。目标语言为西班牙语 (es)。作为并行数据的输入文件，该文件提供了几个可供 Amazon Translate 用于定制批处理任务输出的示例。

Example CSV 输入文件

```
en,es
Amazon Translate is a neural machine translation service.,Amazon Translate es un
 servicio de traducción automática basado en redes neuronales.
Neural machine translation is a form of language translation automation that uses
 deep learning models.,La traducción automática neuronal es una forma de automatizar
 la traducción de lenguajes utilizando modelos de aprendizaje profundo.
Amazon Translate allows you to localize content for international users.,Amazon
 Translate le permite localizar contenido para usuarios internacionales.
```

CSV 要求

在 CSV 文件中定义并行数据时，请记住 Amazon Translate 的以下要求：

- 第一行包含语言代码。第一个代码是源语言，随后的每个代码都是目标语言。
- 第一列中的每个字段都包含源文本。后续列中的每个字段都包含目标翻译。
- 如果任何字段中的文本包含逗号，则该文本必须用双引号 (") 字符括起来。
- 一个文本字段不能跨越多行。
- 字段不能以以下字符开头：+、-、=、@。无论该字段是否用双引号 (") 括起来，此要求都适用。
- 如果字段中的文本包含双引号 (")，则必须使用双引号对其进行转义。例如，以下文本：

  ```
  34" monitor
  ```

必须写成：

```
34"" monitor
```

- 在处理您的输入文件时，如果遇到空白或仅包含空格的字段，Amazon Translate 将跳过某些行或字段：
  - 如果源文本字段为空，Amazon Translate 会跳过该字段所占用的行。
  - 如果目标翻译字段为空，Amazon Translate 将仅跳过该字段。
- 在处理您的输入文件时，如果遇到超过 1000 字节的字段，Amazon Translate 会跳过某些行或字段：
  - 如果源文本字段超出字节限制，Amazon Translate 会跳过该字段所占用的行。
  - 如果目标翻译字段超出字节限制，Amazon Translate 将仅跳过该字段。
- 如果输入文件包含多条具有相同源文本的记录，Amazon Translate 将使用最接近文件末尾的记录。

TSV

以下示例 TSV 文件以 Amazon Translate 可接受的格式定义并行数据。在此文件中，英语 (en) 是源语言。目标语言为西班牙语 (es)。作为并行数据的输入文件，该文件提供了几个可供 Amazon Translate 用于定制批处理任务输出的示例。

Example TSV 输入文件

```
en es
Amazon Translate is a neural machine translation service. Amazon Translate es un
 servicio de traducción automática basado en redes neuronales.
Neural machine translation is a form of language translation automation that uses
 deep learning models. La traducción automática neuronal es una forma de automatizar
 la traducción de lenguajes utilizando modelos de aprendizaje profundo.
Amazon Translate allows you to localize content for international users. Amazon
 Translate le permite localizar contenido para usuarios internacionales.
```

CSV 要求

在 TSV 文件中定义并行数据时，请记住 Amazon Translate 的以下要求：

- 第一行包含语言代码。第一个代码是源语言，随后的每个代码都是目标语言。
- 第一列中的每个字段都包含源文本。后续列中的每个字段都包含目标翻译。

- 如果任何字段中的文本包含制表符，则该文本必须用双引号 (") 字符括起来。
- 一个文本字段不能跨越多行。
- 字段不能以以下字符开头：+、-、=、@。无论该字段是否用双引号 (") 括起来，此要求都适用。
- 如果字段中的文本包含双引号 (")，则必须使用双引号对其进行转义。例如，以下文本：

  ```
  34" monitor
  ```

  必须写成：

  ```
  34"" monitor
  ```

- 在处理您的输入文件时，如果遇到空白或仅包含空格的字段，Amazon Translate 会跳过某些行或字段：
  - 如果源文本字段为空，Amazon Translate 会跳过该字段所占用的行。
  - 如果目标翻译字段为空，Amazon Translate 将仅跳过该字段。
- 在处理您的输入文件时，如果遇到超过 1000 字节的字段，Amazon Translate 会跳过某些行或字段：
  - 如果源文本字段超出字节限制，Amazon Translate 会跳过该字段所占用的行。
  - 如果目标翻译字段超出字节限制，Amazon Translate 将仅跳过该字段。
- 如果输入文件包含多条具有相同源文本的记录，Amazon Translate 将使用最接近文件末尾的记录。

将您的并行数据添加到 Amazon Translate

要向 Amazon Translate 添加并行数据，您需要从 Amazon S3 导入并行数据输入文件。之后，您可以使用并行数据来自定义批量翻译作业生成的输出。

> **先决条件**
>
> 在向 Amazon Translate 添加并行数据之前，您必须：
>
> - 有一个并行数据输入文件。要创建该文件，请参阅 Amazon Translate 的并行数据输入文件。
> - 您的 AWS 账户中有一个 Amazon S3 存储桶。要创建存储桶，请参阅《Amazon Simple Storage Service 用户指南》中的如何创建 S3 存储桶？。

- 将输入文件上传到一个 Amazon S3 存储桶。有关更多信息，请参阅《Amazon Simple Storage Service 用户指南》中的如何将文件和文件夹上传到 S3 存储桶。

添加并行数据（Amazon Translate 控制台）

要使用 Amazon Translate 控制台添加并行数据，请使用并行数据页面：

1. 打开 Amazon Translate 控制台。
2. 在左侧导航菜单中，选择自定义，然后选择并行数据。
3. 在并行数据页面上，选择创建并行数据。控制台显示创建并行数据页面。
4. 提供以下项：

   名称

   并行数据资源的自定义名称。您必须分配一个在账户和区域中唯一的名称。

   描述 - 可选

   自定义描述。

   S3 上的并行数据位置

   Amazon S3 中并行数据输入文件的位置。要通过导航到 Amazon S3 中的文件来提供位置，请选择选择文件。

   文件格式

   并行数据输入文件的格式。支持的格式包括翻译记忆库交换 (TMX)、逗号分隔值 (CSV) 和制表符分隔值 (TSV)。

5. 在加密密钥下，选择一个 AWS KMS key 来保护您的并行数据。这些 KMS 密钥由 AWS Key Management Service (AWS KMS) 管理。有关的更多信息 AWS KMS，请参阅《AWS Key Management Service 开发人员指南》。

   使用 AWS 拥有的密钥

   使用 Amazon Translate 拥有和管理的 KMS 密钥。这是默认选项，如果您不选择其他方法，则此方法用于加密您的信息。有关更多信息，请参阅《AWS Key Management Service 开发人员指南》中的 AWS 拥有的密钥。

使用当前账户中的密钥

使用您在 AWS 账户 AWS KMS 中管理的 KMS 密钥之一。如果您选择此选项，会有一个菜单提供您的 KMS 密钥列表以供选择。有关更多信息，请参阅《AWS Key Management Service 开发人员指南》中的客户托管密钥。

使用不同账户中的密钥

使用在其他 AWS 账户 AWS KMS 中管理的 KMS 密钥。如果选择此选项，控制台中将显示一个字段，供您输入 KMS 密钥的 Amazon 资源名称 (ARN)。

有关加密密钥的更多信息，请参阅《AWS Key Management Service 开发人员指南》。

6. 选择创建并行数据。

   控制台返回并行数据页面，其中的导入状态显示在页面顶部的横幅中。几分钟后，您的并行数据资源将显示在表中。当状态列中的值为活跃时，并行数据即准备好在批量翻译作业中使用。

用于故障排除的错误文件

如果 Amazon Translate 在处理您的输入文件时生成任何错误或警告，控制台会提供一个错误文件，您可以下载该文件来查看错误消息。以下是与该文件内容类似的示例：

```
{
  "summary": {
    "record_error_count": 1,
    "record_skipped_count": 0
  },
  "messages": [
    {
      "content": "Number 1 TU element",
      "message": "Invalid TMX format. One tu element should contain exactly one tuv
 element with the source language code: en"
    }
  ]
}
```

添加并行数据 (AWS CLI)

要使用添加并行数据 AWS CLI，请使用`create-parallel-data`命令。

Example create-parallel-data 命令

以下示例通过从 Amazon S3 导入 TSV 文件来创建并行数据对象：

```
$ aws translate create-parallel-data \
> --name my-parallel-data \
> --parallel-data-config S3Uri=s3://input-bucket/parallel-data-file.tsv,Format=TSV
```

如果命令成功，Amazon Translate 将返回新并行数据对象的状态：

```
{
    "Name": "my-parallel-data",
    "Status": "CREATING"
}
```

您可以使用 `get-parallel-data` 命令监控并行数据的实时状态。当状态为 `ACTIVE` 时，并行数据即准备好在批量翻译作业中使用。有关 `get-parallel-data` 命令的示例，请参阅 查看并行数据对象的详细信息。

使用并行数据

现在，您已创建并行数据资源，可以将其应用于批量翻译作业以自定义输出。要运行批处理作业，请参阅运行批量翻译作业。

在 Amazon Translate 中查看和管理您的并行数据

您可以查看已添加到 Amazon Translate 的所有并行数据资源，也可以访问每个资源的详细摘要。随着翻译要求的变化，您可以通过更新并行数据来优化翻译输出。

查看和管理并行数据（Amazon Translate 控制台）

要在 Amazon Translate 控制台中添加并行数据，请使用并行数据页面：

查看您的并行数据资源列表

1. 打开 Amazon Translate 控制台。
2. 在左侧导航菜单中，选择自定义，然后选择并行数据。本页的表列出了您已添加到 Amazon Translate 的并行数据资源。

查看并行数据资源的详细信息

- 在并行数据页面上，在名称列中选择并行数据资源的名称。控制台会打开详细信息页面，其中包含状态、上次更新日期、源语言和目标语言等信息。

更新并行数据资源

1. 将并行数据的更新版本作为新输入文件上传到 Amazon S3 存储桶中。
2. 在 Amazon Translate 控制台中，转到并行数据页面。
3. 选择想要更新的并行数据，然后选择更新。控制台显示创建并行数据页面。
4. 提供以下项：

   描述 - 可选

   更新的描述。

   S3 上的并行数据位置

   Amazon S3 中更新的并行数据输入文件的位置。要通过导航到 Amazon S3 中的文件来提供位置，请选择选择文件。

   选择并行数据文件格式

   并行数据输入文件的格式。支持的格式包括翻译记忆库交换 (TMX)、逗号分隔值 (CSV) 和制表符分隔值 (TSV)。
5. 选择保存。Amazon Translate 会将旧的并行数据替换为新的输入文件。

查看和管理并行数据 (AWS CLI)

您可以使用 AWS CLI 来查看和更新您的并行数据资源。

查看您的并行数据资源列表

要查看您已添加到 Amazon Translate 的并行数据资源列表，请使用 `list-parallel-data` 命令。

Example list-parallel-data 命令

以下示例返回并行数据资源及其属性的列表。

```
$ aws translate list-parallel-data
```

如果此命令成功，Amazon Translate 将返回类似以下内容的数组：

```
{
    "ParallelDataPropertiesList": [
        {
            "Name": "my-parallel-data",
            "Arn": "arn:aws:translate:us-west-2:111122223333:parallel-data/my-parallel-
data",
            "Status": "ACTIVE",
            "SourceLanguageCode": "en",
            "TargetLanguageCodes": [
                "es",
                "ja",
                "zh"
            ],
            "ParallelDataConfig": {
                "S3Uri": "s3://input-bucket/parallel-data-file.tsv",
                "Format": "TSV"
            },
            "ImportedDataSize": 2283,
            "ImportedRecordCount": 3,
            "FailedRecordCount": 0,
            "CreatedAt": 1598597751.406,
            "LastUpdatedAt": 1598597911.675
        }
    ]
}
```

查看并行数据对象的详细信息

要查找单个并行数据资源的详细信息，请使用 `get-parallel-data` 命令。此命令将返回并行数据的属性以及预签名 S3 URL，您可以通过该 URL 下载用于创建该数据的输入文件。

Example get-parallel-data 命令

以下示例可获取 `my-parallel-data` 对象的属性和下载位置：

```
$ aws translate get-parallel-data \
> --name my-parallel-data
```

如果命令成功，Amazon Translate 将返回属性和下载位置：

```
{
```

```
    "ParallelDataProperties": {
        "Name": "my-parallel-data",
        "Arn": "arn:aws:translate:us-west-2:111122223333:parallel-data/my-parallel-
data",
        "Status": "ACTIVE",
        "SourceLanguageCode": "en",
        "TargetLanguageCodes": [
            "es",
            "ja",
            "zh"
        ],
        "ParallelDataConfig": {
            "S3Uri": "s3://input-bucket/parallel-data-file.tsv",
            "Format": "TSV"
        },
        "ImportedDataSize": 2283,
        "ImportedRecordCount": 3,
        "FailedRecordCount": 0,
        "CreatedAt": 1598597751.406,
        "LastUpdatedAt": 1598597911.675
    },
    "DataLocation": {
        "RepositoryType": "S3",
        "Location": "pre-signed S3 URL"
    }
}
```

更新并行数据资源

要更新并行数据资源，请先将新的输入文件上传到 Amazon S3 输入存储桶。然后，使用 `update-parallel-data` 命令并指定想要更新的并行数据资源。Amazon Translate 将使用新输入文件中的信息替换旧的并行数据。

Example update-parallel-data 命令

以下命令使用来自 Amazon S3 的新输入文件更新 `my-parallel-data`：

```
$ aws translate update-parallel-data \
> --name my-parallel-data \
> --parallel-data-config S3Uri=s3://input-bucket/parallel-data-file.tsv,Format=TSV
```

如果此命令成功，Amazon Translate 将返回如下响应：

```
{
    "Name": "my-parallel-data",
    "Status": "ACTIVE",
    "LatestUpdateAttemptStatus": "UPDATING",
    "LatestUpdateAttemptAt": 1598601455.844
}
```

在此响应中，Status 字段提供预先存在的并行数据对象的状态，LatestUpdateAttemptStatus 字段提供当前更新尝试的状态。

使用 Amazon Translate 的代码示例 AWS SDKs

以下代码示例展示了如何将 Amazon Translate 与 AWS 软件开发套件 (SDK) 一起使用。

操作是大型程序的代码摘录，必须在上下文中运行。您可以通过操作了解如何调用单个服务函数，还可以通过函数相关场景的上下文查看操作。

场景是向您展示如何通过在一个服务中调用多个函数或与其他 AWS 服务服务结合来完成特定任务的代码示例。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

代码示例

- 使用 Amazon Translate 的基本示例 AWS SDKs
  - 使用 Amazon Translate 执行的操作 AWS SDKs
    - 与 AWS SDK DescribeTextTranslationJob 配合使用
    - 与 AWS SDK ListTextTranslationJobs 配合使用
    - 与 AWS SDK StartTextTranslationJob 配合使用
    - 与 AWS SDK StopTextTranslationJob 配合使用
    - TranslateText与 AWS SDK 或 CLI 配合使用
- Amazon Translate 使用场景 AWS SDKs
  - 构建 Amazon Transcribe 流式传输应用程序
  - 创建 Amazon Lex 聊天机器人来吸引您的网站访客
  - 构建转换消息的发布和订阅应用程序
  - 创建用于分析客户反馈和合成音频的应用程序
  - 使用 AWS 软件开发工具包开始使用 Amazon Translate 作业

使用 Amazon Translate 的基本示例 AWS SDKs

以下代码示例展示了如何使用 Amazon Translate 的基础知识 AWS SDKs。

示例

- 使用 Amazon Translate 执行的操作 AWS SDKs

- 与 AWS SDK DescribeTextTranslationJob 配合使用
- 与 AWS SDK ListTextTranslationJobs 配合使用
- 与 AWS SDK StartTextTranslationJob 配合使用
- 与 AWS SDK StopTextTranslationJob 配合使用
- TranslateText与 AWS SDK 或 CLI 配合使用

使用 Amazon Translate 执行的操作 AWS SDKs

以下代码示例演示了如何使用执行各个 Amazon Translate 操作 AWS SDKs。每个示例都包含一个指向的链接 GitHub，您可以在其中找到有关设置和运行代码的说明。

这些摘录调用 Amazon Translate API，是大型程序的代码摘录，这些程序必须在上下文中运行。您可以在Amazon Translate 使用场景 AWS SDKs 中结合上下文查看操作。

以下示例仅包括最常用的操作。如需完整列表，请参阅 Amazon Translate API 参考。

示例

- 与 AWS SDK DescribeTextTranslationJob 配合使用
- 与 AWS SDK ListTextTranslationJobs 配合使用
- 与 AWS SDK StartTextTranslationJob 配合使用
- 与 AWS SDK StopTextTranslationJob 配合使用
- TranslateText与 AWS SDK 或 CLI 配合使用

与 AWS SDK **`DescribeTextTranslationJob`** 配合使用

以下代码示例演示如何使用 `DescribeTextTranslationJob`。

操作示例是大型程序的代码摘录，必须在上下文中运行。在以下代码示例中，您可以查看此操作的上下文：

- 开始使用翻译任务

.NET

适用于 .NET 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
using System;
using System.Threading.Tasks;
using Amazon.Translate;
using Amazon.Translate.Model;

/// <summary>
/// The following example shows how to retrieve the details of
/// a text translation job using Amazon Translate.
/// </summary>
public class DescribeTextTranslation
{
    public static async Task Main()
    {
        var client = new AmazonTranslateClient();

        // The Job Id is generated when the text translation job is started
        // with a call to the StartTextTranslationJob method.
        var jobId = "1234567890abcdef01234567890abcde";

        var request = new DescribeTextTranslationJobRequest
        {
            JobId = jobId,
        };

        var jobProperties = await DescribeTranslationJobAsync(client,
request);

        DisplayTranslationJobDetails(jobProperties);
    }

    /// <summary>
    /// Retrieve information about an Amazon Translate text translation job.
```

```
        /// </summary>
        /// <param name="client">The initialized Amazon Translate client
 object.</param>
        /// <param name="request">The DescribeTextTranslationJobRequest object.</
param>
        /// <returns>The TextTranslationJobProperties object containing
        /// information about the text translation job..</returns>
        public static async Task<TextTranslationJobProperties>
 DescribeTranslationJobAsync(
            AmazonTranslateClient client,
            DescribeTextTranslationJobRequest request)
        {
            var response = await client.DescribeTextTranslationJobAsync(request);
            if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
            {
                return response.TextTranslationJobProperties;
            }
            else
            {
                return null;
            }
        }

        /// <summary>
        /// Displays the properties of the text translation job.
        /// </summary>
        /// <param name="jobProperties">The properties of the text translation
        /// job returned by the call to DescribeTextTranslationJobAsync.</param>
        public static void
 DisplayTranslationJobDetails(TextTranslationJobProperties jobProperties)
        {
            if (jobProperties is null)
            {
                Console.WriteLine("No text translation job properties found.");
                return;
            }

            // Display the details of the text translation job.
            Console.WriteLine($"{jobProperties.JobId}: {jobProperties.JobName}");
        }
    }
```

- 有关 API 的详细信息，请参阅 适用于 .NET 的 AWS SDK API 参考DescribeTextTranslationJob中的。

SAP ABAP

适用于 SAP ABAP 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
   "Gets the properties associated with an asynchronous batch translation job."
   "Includes properties such as name, ID, status, source and target languages,
and input/output Amazon Simple Storage Service (Amazon S3) buckets."
   TRY.
       oo_result = lo_xl8->describetexttranslationjob(      "oo_result is
returned for testing purposes."
         iv_jobid        = iv_jobid ).
       MESSAGE 'Job description retrieved.' TYPE 'I'.
     CATCH /aws1/cx_xl8internalserverex.
       MESSAGE 'An internal server error occurred. Retry your request.' TYPE
'E'.
     CATCH /aws1/cx_xl8resourcenotfoundex.
       MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
     CATCH /aws1/cx_xl8toomanyrequestsex.
       MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
   ENDTRY.
```

- 有关 API 的详细信息，请参阅适用DescribeTextTranslationJob于 S AP 的AWS SDK ABAP API 参考。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

与 AWS SDK `ListTextTranslationJobs` 配合使用

以下代码示例演示如何使用 `ListTextTranslationJobs`。

.NET

适用于 .NET 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
using System;
using System.Collections.Generic;
using System.Threading.Tasks;
using Amazon.Translate;
using Amazon.Translate.Model;

/// <summary>
/// List Amazon Translate translation jobs, along with details about each
job.
/// </summary>
public class ListTranslationJobs
{
    public static async Task Main()
    {
        var client = new AmazonTranslateClient();
        var filter = new TextTranslationJobFilter
        {
            JobStatus = "COMPLETED",
        };

        var request = new ListTextTranslationJobsRequest
        {
            MaxResults = 10,
            Filter = filter,
        };

        await ListJobsAsync(client, request);
    }
```

```
        /// <summary>
        /// List Amazon Translate text translation jobs.
        /// </summary>
        /// <param name="client">The initialized Amazon Translate client
object.</param>
        /// <param name="request">An Amazon Translate
        /// ListTextTranslationJobsRequest object detailing which text
        /// translation jobs are of interest.</param>
        public static async Task ListJobsAsync(
            AmazonTranslateClient client,
            ListTextTranslationJobsRequest request)
        {
            ListTextTranslationJobsResponse response;

            do
            {
                response = await client.ListTextTranslationJobsAsync(request);

ShowTranslationJobDetails(response.TextTranslationJobPropertiesList);

                request.NextToken = response.NextToken;
            }
            while (response.NextToken is not null);
        }

        /// <summary>
        /// List existing translation job details.
        /// </summary>
        /// <param name="properties">A list of Amazon Translate text
        /// translation jobs.</param>
        public static void
ShowTranslationJobDetails(List<TextTranslationJobProperties> properties)
        {
            properties.ForEach(prop =>
            {
                Console.WriteLine($"{prop.JobId}: {prop.JobName}");
                Console.WriteLine($"Status: {prop.JobStatus}");
                Console.WriteLine($"Submitted time: {prop.SubmittedTime}");
            });
        }
    }
```

- 有关 API 的详细信息，请参阅 适用于 .NET 的 AWS SDK API 参考ListTextTranslationJobs中的。

SAP ABAP

适用于 SAP ABAP 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
   "Gets a list of the batch translation jobs that you have submitted."

   DATA lo_filter TYPE REF TO /aws1/cl_xl8textxlationjobfilt.

   "Create an ABAP object for filtering using jobname."
   lo_filter = NEW #( iv_jobname = iv_jobname ).

   TRY.
       oo_result = lo_xl8->listtexttranslationjobs(      "oo_result is returned
for testing purposes."
         io_filter        = lo_filter ).
       MESSAGE 'Jobs retrieved.' TYPE 'I'.
     CATCH /aws1/cx_xl8internalserverex.
       MESSAGE 'An internal server error occurred. Retry your request.' TYPE
'E'.
     CATCH /aws1/cx_xl8invalidfilterex.
       MESSAGE 'The filter specified for the operation is not valid. Specify a
different filter.' TYPE 'E'.
     CATCH /aws1/cx_xl8invalidrequestex.
       MESSAGE 'The request that you made is not valid.' TYPE 'E'.
     CATCH /aws1/cx_xl8toomanyrequestsex.
       MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
   ENDTRY.
```

- 有关 API 的详细信息，请参阅适用ListTextTranslationJobs于 S AP 的AWS SDK ABAP API 参考。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

与 AWS SDK **StartTextTranslationJob** 配合使用

以下代码示例演示如何使用 `StartTextTranslationJob`。

操作示例是大型程序的代码摘录，必须在上下文中运行。在以下代码示例中，您可以查看此操作的上下文：

- 开始使用翻译任务

.NET

适用于 .NET 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
    using System;
    using System.Collections.Generic;
    using System.Threading.Tasks;
    using Amazon.Translate;
    using Amazon.Translate.Model;

    /// <summary>
    /// This example shows how to use Amazon Translate to process the files in
    /// an Amazon Simple Storage Service (Amazon S3) bucket. The translated
results
    /// will also be stored in an Amazon S3 bucket.
    /// </summary>
    public class BatchTranslate
    {
        public static async Task Main()
```

```
        {
            var contentType = "text/plain";

            // Set this variable to an S3 bucket location with a folder."
            // Input files must be in a folder and not at the bucket root."
            var s3InputUri = "s3://amzn-s3-demo-bucket1/FOLDER/";
            var s3OutputUri = "s3://amzn-s3-demo-bucket2/";

            // This role must have permissions to read the source bucket and to
 read and
            // write to the destination bucket where the translated text will be
 stored.
            var dataAccessRoleArn = "arn:aws:iam::0123456789ab:role/
S3TranslateRole";

            var client = new AmazonTranslateClient();

            var inputConfig = new InputDataConfig
            {
                ContentType = contentType,
                S3Uri = s3InputUri,
            };

            var outputConfig = new OutputDataConfig
            {
                S3Uri = s3OutputUri,
            };

            var request = new StartTextTranslationJobRequest
            {
                JobName = "ExampleTranslationJob",
                DataAccessRoleArn = dataAccessRoleArn,
                InputDataConfig = inputConfig,
                OutputDataConfig = outputConfig,
                SourceLanguageCode = "en",
                TargetLanguageCodes = new List<string> { "fr" },
            };

            var response = await StartTextTranslationAsync(client, request);

            if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
            {
                Console.WriteLine($"{response.JobId}: {response.JobStatus}");
            }
```

```
        }

        /// <summary>
        /// Start the Amazon Translate text translation job.
        /// </summary>
        /// <param name="client">The initialized AmazonTranslateClient object.</
param>
        /// <param name="request">The request object that includes details such
        /// as source and destination bucket names and the IAM Role that will
        /// be used to access the buckets.</param>
        /// <returns>The StartTextTranslationResponse object that includes the
        /// details of the request response.</returns>
        public static async Task<StartTextTranslationJobResponse>
 StartTextTranslationAsync(AmazonTranslateClient client,
 StartTextTranslationJobRequest request)
        {
            var response = await client.StartTextTranslationJobAsync(request);
            return response;
        }
    }
```

- 有关 API 的详细信息，请参阅 适用于 .NET 的 AWS SDK API 参考StartTextTranslationJob中的。

SAP ABAP

适用于 SAP ABAP 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
    "Starts an asynchronous batch translation job."
    "Use batch translation jobs to translate large volumes of text across
 multiple documents at once."

    DATA lo_inputdataconfig  TYPE REF TO /aws1/cl_xl8inputdataconfig.
```

```
    DATA lo_outputdataconfig TYPE REF TO /aws1/cl_xl8outputdataconfig.
    DATA lt_targetlanguagecodes TYPE /aws1/
cl_xl8tgtlanguagecodes00=>tt_targetlanguagecodestrlist.
    DATA lo_targetlanguagecodes TYPE REF TO /aws1/cl_xl8tgtlanguagecodes00.

    "Create an ABAP object for the input data config."
    lo_inputdataconfig = NEW #( iv_s3uri = iv_input_data_s3uri
                                iv_contenttype = iv_input_data_contenttype ).

    "Create an ABAP object for the output data config."
    lo_outputdataconfig = NEW #( iv_s3uri = iv_output_data_s3uri ).

    "Create an internal table for target languages."
    lo_targetlanguagecodes = NEW #( iv_value = iv_targetlanguagecode ).
    INSERT lo_targetlanguagecodes  INTO TABLE lt_targetlanguagecodes.

    TRY.
        oo_result = lo_xl8->starttexttranslationjob(      "oo_result is returned
 for testing purposes."
          io_inputdataconfig = lo_inputdataconfig
            io_outputdataconfig = lo_outputdataconfig
            it_targetlanguagecodes = lt_targetlanguagecodes
            iv_dataaccessrolearn = iv_dataaccessrolearn
            iv_jobname = iv_jobname
            iv_sourcelanguagecode = iv_sourcelanguagecode ).
        MESSAGE 'Translation job started.' TYPE 'I'.
      CATCH /aws1/cx_xl8internalserverex.
        MESSAGE 'An internal server error occurred. Retry your request.' TYPE
 'E'.
      CATCH /aws1/cx_xl8invparamvalueex.
        MESSAGE 'The value of the parameter is not valid.' TYPE 'E'.
      CATCH /aws1/cx_xl8invalidrequestex.
        MESSAGE 'The request that you made is not valid.' TYPE 'E'.
      CATCH /aws1/cx_xl8resourcenotfoundex.
        MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
      CATCH /aws1/cx_xl8toomanyrequestsex.
        MESSAGE 'You have made too many requests within a short period of time.'
 TYPE 'E'.
      CATCH /aws1/cx_xl8unsuppedlanguage00.
        MESSAGE 'Amazon Translate does not support translation from the language
 of the source text into the requested target language.' TYPE 'E'.
    ENDTRY.
```

- 有关 API 的详细信息，请参阅适用StartTextTranslationJob于 S AP 的AWS SDK ABAP API 参考。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

与 AWS SDK **StopTextTranslationJob** 配合使用

以下代码示例演示如何使用 `StopTextTranslationJob`。

.NET

适用于 .NET 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
using System;
using System.Threading.Tasks;
using Amazon.Translate;
using Amazon.Translate.Model;

/// <summary>
/// Shows how to stop a running Amazon Translation Service text translation
/// job.
/// </summary>
public class StopTextTranslationJob
{
    public static async Task Main()
    {
        var client = new AmazonTranslateClient();
        var jobId = "1234567890abcdef01234567890abcde";

        var request = new StopTextTranslationJobRequest
        {
            JobId = jobId,
        };
```

```
            await StopTranslationJobAsync(client, request);
        }

        /// <summary>
        /// Sends a request to stop a text translation job.
        /// </summary>
        /// <param name="client">Initialized AmazonTrnslateClient object.</param>
        /// <param name="request">The request object to be passed to the
        /// StopTextJobAsync method.</param>
        public static async Task StopTranslationJobAsync(
            AmazonTranslateClient client,
            StopTextTranslationJobRequest request)
        {
            var response = await client.StopTextTranslationJobAsync(request);
            if (response.HttpStatusCode == System.Net.HttpStatusCode.OK)
            {
                Console.WriteLine($"{response.JobId} as status:
{response.JobStatus}");
            }
        }
    }
```

- 有关 API 的详细信息，请参阅 适用于 .NET 的 AWS SDK API 参考StopTextTranslationJob中的。

SAP ABAP

适用于 SAP ABAP 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
    "Stops an asynchronous batch translation job that is in progress."

    TRY.
```

```
        oo_result = lo_xl8->stoptexttranslationjob(      "oo_result is returned
for testing purposes."
          iv_jobid        = iv_jobid ).
        MESSAGE 'Translation job stopped.' TYPE 'I'.
      CATCH /aws1/cx_xl8internalserverex.
        MESSAGE 'An internal server error occurred.' TYPE 'E'.
      CATCH /aws1/cx_xl8resourcenotfoundex.
        MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
      CATCH /aws1/cx_xl8toomanyrequestsex.
        MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
    ENDTRY.
```

- 有关 API 的详细信息，请参阅适用StopTextTranslationJob于 S AP 的AWS SDK ABAP API 参考。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

`TranslateText`与 AWS SDK 或 CLI 配合使用

以下代码示例演示如何使用 `TranslateText`。

.NET

适用于 .NET 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
using System;
using System.IO;
using System.Threading.Tasks;
using Amazon.S3;
using Amazon.S3.Transfer;
using Amazon.Translate;
using Amazon.Translate.Model;
```

```
    /// <summary>
    /// Take text from a file stored a Amazon Simple Storage Service (Amazon S3)
    /// object and translate it using the Amazon Transfer Service.
    /// </summary>
    public class TranslateText
    {
        public static async Task Main()
        {
            // If the region you want to use is different from the region
            // defined for the default user, supply it as a parameter to the
            // Amazon Translate client object constructor.
            var client = new AmazonTranslateClient();

            // Set the source language to "auto" to request Amazon Translate to
            // automatically detect te language of the source text.

            // You can get a list of the languages supposed by Amazon Translate
            // in the Amazon Translate Developer's Guide here:
            //      https://docs.aws.amazon.com/translate/latest/dg/what-is.html
            string srcLang = "en"; // English.
            string destLang = "fr"; // French.

            // The Amazon Simple Storage Service (Amazon S3) bucket where the
            // source text file is stored.
            string srcBucket = "amzn-s3-demo-bucket";
            string srcTextFile = "source.txt";

            var srcText = await GetSourceTextAsync(srcBucket, srcTextFile);
            var destText = await TranslatingTextAsync(client, srcLang, destLang,
srcText);

            ShowText(srcText, destText);
        }

        /// <summary>
        /// Use the Amazon S3 TransferUtility to retrieve the text to translate
        /// from an object in an S3 bucket.
        /// </summary>
        /// <param name="srcBucket">The name of the S3 bucket where the
        /// text is stored.
        /// </param>
        /// <param name="srcTextFile">The key of the S3 object that
        /// contains the text to translate.</param>
```

```
        /// <returns>A string representing the source text.</returns>
        public static async Task<string> GetSourceTextAsync(string srcBucket,
 string srcTextFile)
        {
            string srcText = string.Empty;

            var s3Client = new AmazonS3Client();
            TransferUtility utility = new TransferUtility(s3Client);

            using var stream = await utility.OpenStreamAsync(srcBucket,
 srcTextFile);

            StreamReader file = new System.IO.StreamReader(stream);

            srcText = file.ReadToEnd();
            return srcText;
        }

        /// <summary>
        /// Use the Amazon Translate Service to translate the document from the
        /// source language to the specified destination language.
        /// </summary>
        /// <param name="client">The Amazon Translate Service client used to
        /// perform the translation.</param>
        /// <param name="srcLang">The language of the source text.</param>
        /// <param name="destLang">The destination language for the translated
        /// text.</param>
        /// <param name="text">A string representing the text to ranslate.</
param>
        /// <returns>The text that has been translated to the destination
        /// language.</returns>
        public static async Task<string>
 TranslatingTextAsync(AmazonTranslateClient client, string srcLang, string
 destLang, string text)
        {
            var request = new TranslateTextRequest
            {
                SourceLanguageCode = srcLang,
                TargetLanguageCode = destLang,
                Text = text,
            };

            var response = await client.TranslateTextAsync(request);
```

```
            return response.TranslatedText;
        }

        /// <summary>
        /// Show the original text followed by the translated text.
        /// </summary>
        /// <param name="srcText">The original text to be translated.</param>
        /// <param name="destText">The translated text.</param>
        public static void ShowText(string srcText, string destText)
        {
            Console.WriteLine("Source text:");
            Console.WriteLine(srcText);
            Console.WriteLine();
            Console.WriteLine("Translated text:");
            Console.WriteLine(destText);
        }
    }
```

- 有关 API 的详细信息，请参阅 适用于 .NET 的 AWS SDK API 参考TranslateText中的。

PowerShell

适用于 PowerShell V4 的工具

示例 1：将指定的英文文本转换为法语。要转换的文本也可以作为-Text参数传递。

```
"Hello World" | ConvertTo-TRNTargetLanguage -SourceLanguageCode en -
TargetLanguageCode fr
```

- 有关 API 的详细信息，请参阅 AWS Tools for PowerShell Cmdlet 参考 (V 4) TranslateText中的。

SAP ABAP

适用于 SAP ABAP 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
   "Translates input text from the source language to the target language."
   TRY.
       oo_result = lo_xl8->translatetext(      "oo_result is returned for
testing purposes."
         iv_text        = iv_text
           iv_sourcelanguagecode = iv_sourcelanguagecode
           iv_targetlanguagecode = iv_targetlanguagecode ).
       MESSAGE 'Translation completed.' TYPE 'I'.
     CATCH /aws1/cx_xl8detectedlanguage00.
       MESSAGE 'The confidence that Amazon Comprehend accurately detected the
source language is low.' TYPE 'E'.
     CATCH /aws1/cx_xl8internalserverex.
       MESSAGE 'An internal server error occurred.' TYPE 'E'.
     CATCH /aws1/cx_xl8invalidrequestex.
       MESSAGE 'The request that you made is not valid.' TYPE 'E'.
     CATCH /aws1/cx_xl8resourcenotfoundex.
       MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
     CATCH /aws1/cx_xl8serviceunavailex.
       MESSAGE 'The Amazon Translate service is temporarily unavailable.' TYPE
'E'.
     CATCH /aws1/cx_xl8textsizelmtexcdex.
       MESSAGE 'The size of the text you submitted exceeds the size limit. '
TYPE 'E'.
     CATCH /aws1/cx_xl8toomanyrequestsex.
       MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
     CATCH /aws1/cx_xl8unsuppedlanguage00.
       MESSAGE 'Amazon Translate does not support translation from the language
of the source text into the requested target language. ' TYPE 'E'.
   ENDTRY.
```

- 有关 API 的详细信息，请参阅适用TranslateText于 S AP 的AWS SDK ABAP API 参考。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

Amazon Translate 使用场景 AWS SDKs

以下代码示例向您展示了如何使用在 Amazon Translate 中实现常见场景 AWS SDKs。这些场景向您展示了如何通过在 Amazon Translate 中调用多个函数或与其他函数结合使用来完成特定任务 AWS 服务。每个场景都包含完整源代码的链接，您可以在其中找到有关如何设置和运行代码的说明。

场景以中等水平的经验为目标，可帮助您结合具体环境了解服务操作。

示例

- 构建 Amazon Transcribe 流式传输应用程序
- 创建 Amazon Lex 聊天机器人来吸引您的网站访客
- 构建转换消息的发布和订阅应用程序
- 创建用于分析客户反馈和合成音频的应用程序
- 使用 AWS 软件开发工具包开始使用 Amazon Translate 作业

构建 Amazon Transcribe 流式传输应用程序

以下代码示例展示如何构建可实时录制、转录与翻译实时音频，并通过电子邮件发送结果的应用程序。

JavaScript

适用于 JavaScript (v3) 的软件开发工具包

演示了如何使用 Amazon Transcribe 构建可实时录制、转录与翻译实时音频，并通过 Amazon Simple Email Service (Amazon SES) 以电子邮件发送结果的应用程序。

有关如何设置和运行的完整源代码和说明，请参阅上的完整示例GitHub。

本示例中使用的服务

- Amazon Comprehend
- Amazon SES
- Amazon Transcribe

- Amazon Translate

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

创建 Amazon Lex 聊天机器人来吸引您的网站访客

以下代码示例显示如何创建用于吸引网站访客的聊天机器人。

Java

适用于 Java 的 SDK 2.x

展示如何使用 Amazon Lex API 在 Web 应用程序中创建聊天机器人，以吸引网站访客。

有关如何设置和运行的完整源代码和说明，请参阅上的完整示例GitHub。

本示例中使用的服务

- Amazon Comprehend
- Amazon Lex
- Amazon Translate

JavaScript

适用于 JavaScript (v3) 的软件开发工具包

展示如何使用 Amazon Lex API 在 Web 应用程序中创建聊天机器人，以吸引网站访客。

有关如何设置和运行的完整源代码和说明，请参阅 适用于 JavaScript 的 AWS SDK 开发者指南中的构建 Amazon Lex 聊天机器人的完整示例。

本示例中使用的服务

- Amazon Comprehend
- Amazon Lex
- Amazon Translate

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

构建转换消息的发布和订阅应用程序

以下代码示例展示如何创建具有订阅和发布功能并能转换消息的应用程序。

.NET

适用于 .NET 的 SDK

展示如何使用 Amazon Simple Notification Service .NET API 创建具有订阅和发布功能的 Web 应用程序。此外，此示例应用程序还会转换消息。

有关如何设置和运行的完整源代码和说明，请参阅上的完整示例GitHub。

本示例中使用的服务

- Amazon SNS
- Amazon Translate

Java

适用于 Java 的 SDK 2.x

展示如何使用 Amazon Simple Notification Service Java API 创建具有订阅和发布功能的 Web 应用程序。此外，此示例应用程序还会转换消息。

有关如何设置和运行的完整源代码和说明，请参阅上的完整示例GitHub。

有关如何设置和运行使用 Java Async API 的示例的完整源代码和说明，请参阅上的GitHub完整示例。

本示例中使用的服务

- Amazon SNS
- Amazon Translate

Kotlin

适用于 Kotlin 的 SDK

展示如何使用 Amazon SNS Kotlin API 创建具有订阅和发布功能的应用程序。此外，此示例应用程序还会转换消息。

有关如何创建 Web 应用程序的完整源代码和说明，请参阅上的完整示例GitHub。

有关如何创建原生 Android 应用程序的完整源代码和说明，请参阅上的完整示例GitHub。

本示例中使用的服务

- Amazon SNS
- Amazon Translate

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

创建用于分析客户反馈和合成音频的应用程序

以下代码示例显示如何创建应用程序来分析客户意见卡、翻译其母语、确定其情绪并根据译后的文本生成音频文件。

.NET

适用于 .NET 的 SDK

此示例应用程序可分析并存储客户反馈卡。具体来说，它满足了纽约市一家虚构酒店的需求。酒店以实体意见卡的形式收集来自不同语种的客人的反馈。该反馈通过 Web 客户端上传到应用程序中。意见卡图片上传后，将执行以下步骤：

- 使用 Amazon Textract 从图片中提取文本。
- Amazon Comprehend 确定所提取文本的情绪及其语言。
- 使用 Amazon Translate 将所提取文本翻译为英语。
- Amazon Polly 根据所提取文本合成音频文件。

完整的应用程序可使用 AWS CDK 进行部署。有关源代码和部署说明，请参阅中的项目GitHub。

本示例中使用的服务

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract

- Amazon Translate

Java

适用于 Java 的 SDK 2.x

此示例应用程序可分析并存储客户反馈卡。具体来说，它满足了纽约市一家虚构酒店的需求。酒店以实体意见卡的形式收集来自不同语种的客人的反馈。该反馈通过 Web 客户端上传到应用程序中。意见卡图片上传后，将执行以下步骤：

- 使用 Amazon Textract 从图片中提取文本。
- Amazon Comprehend 确定所提取文本的情绪及其语言。
- 使用 Amazon Translate 将所提取文本翻译为英语。
- Amazon Polly 根据所提取文本合成音频文件。

完整的应用程序可使用 AWS CDK 进行部署。有关源代码和部署说明，请参阅中的项目 GitHub。

本示例中使用的服务

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract
- Amazon Translate

JavaScript

适用于 JavaScript (v3) 的软件开发工具包

此示例应用程序可分析并存储客户反馈卡。具体来说，它满足了纽约市一家虚构酒店的需求。酒店以实体意见卡的形式收集来自不同语种的客人的反馈。该反馈通过 Web 客户端上传到应用程序中。意见卡图片上传后，将执行以下步骤：

- 使用 Amazon Textract 从图片中提取文本。
- Amazon Comprehend 确定所提取文本的情绪及其语言。
- 使用 Amazon Translate 将所提取文本翻译为英语。

- Amazon Polly 根据所提取文本合成音频文件。

完整的应用程序可使用 AWS CDK 进行部署。有关源代码和部署说明，请参阅中的项目 GitHub。以下摘录显示了在 Lambda 函数中 适用于 JavaScript 的 AWS SDK 是如何使用的。

```
import {
  ComprehendClient,
  DetectDominantLanguageCommand,
  DetectSentimentCommand,
} from "@aws-sdk/client-comprehend";

/**
 * Determine the language and sentiment of the extracted text.
 *
 * @param {{ source_text: string}} extractTextOutput
 */
export const handler = async (extractTextOutput) => {
  const comprehendClient = new ComprehendClient({});

  const detectDominantLanguageCommand = new DetectDominantLanguageCommand({
    Text: extractTextOutput.source_text,
  });

  // The source language is required for sentiment analysis and
  // translation in the next step.
  const { Languages } = await comprehendClient.send(
    detectDominantLanguageCommand,
  );

  const languageCode = Languages[0].LanguageCode;

  const detectSentimentCommand = new DetectSentimentCommand({
    Text: extractTextOutput.source_text,
    LanguageCode: languageCode,
  });

  const { Sentiment } = await comprehendClient.send(detectSentimentCommand);

  return {
    sentiment: Sentiment,
    language_code: languageCode,
  };
};
```

```
import {
  DetectDocumentTextCommand,
  TextractClient,
} from "@aws-sdk/client-textract";

/**
 * Fetch the S3 object from the event and analyze it using Amazon Textract.
 *
 * @param {import("@types/aws-lambda").EventBridgeEvent<"Object Created">}
 eventBridgeS3Event
 */
export const handler = async (eventBridgeS3Event) => {
  const textractClient = new TextractClient();

  const detectDocumentTextCommand = new DetectDocumentTextCommand({
    Document: {
      S3Object: {
        Bucket: eventBridgeS3Event.bucket,
        Name: eventBridgeS3Event.object,
      },
    },
  });

  // Textract returns a list of blocks. A block can be a line, a page, word, etc.
  // Each block also contains geometry of the detected text.
  // For more information on the Block type, see https://docs.aws.amazon.com/
textract/latest/dg/API_Block.html.
  const { Blocks } = await textractClient.send(detectDocumentTextCommand);

  // For the purpose of this example, we are only interested in words.
  const extractedWords = Blocks.filter((b) => b.BlockType === "WORD").map(
    (b) => b.Text,
  );

  return extractedWords.join(" ");
};
```

```
import { PollyClient, SynthesizeSpeechCommand } from "@aws-sdk/client-polly";
import { S3Client } from "@aws-sdk/client-s3";
import { Upload } from "@aws-sdk/lib-storage";

/**
 * Synthesize an audio file from text.
```

```
 *
 * @param {{ bucket: string, translated_text: string, object: string}}
 sourceDestinationConfig
 */
export const handler = async (sourceDestinationConfig) => {
  const pollyClient = new PollyClient({});

  const synthesizeSpeechCommand = new SynthesizeSpeechCommand({
    Engine: "neural",
    Text: sourceDestinationConfig.translated_text,
    VoiceId: "Ruth",
    OutputFormat: "mp3",
  });

  const { AudioStream } = await pollyClient.send(synthesizeSpeechCommand);

  const audioKey = `${sourceDestinationConfig.object}.mp3`;

  // Store the audio file in S3.
  const s3Client = new S3Client();
  const upload = new Upload({
    client: s3Client,
    params: {
      Bucket: sourceDestinationConfig.bucket,
      Key: audioKey,
      Body: AudioStream,
      ContentType: "audio/mp3",
    },
  });

  await upload.done();
  return audioKey;
};
```

```
import {
  TranslateClient,
  TranslateTextCommand,
} from "@aws-sdk/client-translate";

/**
 * Translate the extracted text to English.
 *
```

```
 * @param {{ extracted_text: string, source_language_code: string}}
 textAndSourceLanguage
 */
export const handler = async (textAndSourceLanguage) => {
  const translateClient = new TranslateClient({});

  const translateCommand = new TranslateTextCommand({
    SourceLanguageCode: textAndSourceLanguage.source_language_code,
    TargetLanguageCode: "en",
    Text: textAndSourceLanguage.extracted_text,
  });

  const { TranslatedText } = await translateClient.send(translateCommand);

  return { translated_text: TranslatedText };
};
```

本示例中使用的服务

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract
- Amazon Translate

Ruby

适用于 Ruby 的 SDK

此示例应用程序可分析并存储客户反馈卡。具体来说，它满足了纽约市一家虚构酒店的需求。酒店以实体意见卡的形式收集来自不同语种的客人的反馈。该反馈通过 Web 客户端上传到应用程序中。意见卡图片上传后，将执行以下步骤：

- 使用 Amazon Textract 从图片中提取文本。
- Amazon Comprehend 确定所提取文本的情绪及其语言。
- 使用 Amazon Translate 将所提取文本翻译为英语。
- Amazon Polly 根据所提取文本合成音频文件。

完整的应用程序可使用 AWS CDK 进行部署。有关源代码和部署说明，请参阅中的项目 GitHub。

本示例中使用的服务

- Amazon Comprehend
- Lambda
- Amazon Polly
- Amazon Textract
- Amazon Translate

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

使用 AWS 软件开发工具包开始使用 Amazon Translate 作业

以下代码示例显示了操作流程：

- 启动异步批处理翻译任务。
- 等待异步任务完成。
- 描述异步任务。

SAP ABAP

适用于 SAP ABAP 的 SDK

> **Note**
>
> 还有更多相关信息 GitHub。在 AWS 代码示例存储库中查找完整示例，了解如何进行设置和运行。

```
    DATA lo_inputdataconfig  TYPE REF TO /aws1/cl_xl8inputdataconfig.
    DATA lo_outputdataconfig TYPE REF TO /aws1/cl_xl8outputdataconfig.
    DATA lt_targetlanguagecodes TYPE /aws1/
cl_xl8tgtlanguagecodes00=>tt_targetlanguagecodestrlist.
    DATA lo_targetlanguagecodes TYPE REF TO /aws1/cl_xl8tgtlanguagecodes00.

    "Create an ABAP object for the input data config."
    lo_inputdataconfig = NEW #( iv_s3uri = iv_input_data_s3uri
```

```
                                iv_contenttype = iv_input_data_contenttype ).

    "Create an ABAP object for the output data config."
    lo_outputdataconfig = NEW #( iv_s3uri = iv_output_data_s3uri ).

    "Create an internal table for target languages."
    lo_targetlanguagecodes = NEW #( iv_value = iv_targetlanguagecode ).
    INSERT lo_targetlanguagecodes  INTO TABLE lt_targetlanguagecodes.

    TRY.
        DATA(lo_translationjob_result) = lo_xl8->starttexttranslationjob(
          io_inputdataconfig = lo_inputdataconfig
            io_outputdataconfig = lo_outputdataconfig
            it_targetlanguagecodes = lt_targetlanguagecodes
            iv_dataaccessrolearn = iv_dataaccessrolearn
            iv_jobname = iv_jobname
            iv_sourcelanguagecode = iv_sourcelanguagecode ).
        MESSAGE 'Translation job started.' TYPE 'I'.
      CATCH /aws1/cx_xl8internalserverex.
        MESSAGE 'An internal server error occurred. Retry your request.' TYPE
 'E'.
      CATCH /aws1/cx_xl8invparamvalueex.
        MESSAGE 'The value of the parameter is not valid.' TYPE 'E'.
      CATCH /aws1/cx_xl8invalidrequestex.
        MESSAGE 'The request that you made is not valid.' TYPE 'E'.
      CATCH /aws1/cx_xl8resourcenotfoundex.
        MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
      CATCH /aws1/cx_xl8toomanyrequestsex.
        MESSAGE 'You have made too many requests within a short period of time. '
 TYPE 'E'.
      CATCH /aws1/cx_xl8unsuppedlanguage00.
        MESSAGE 'Amazon Translate does not support translation from the language
 of the source text into the requested target language.' TYPE 'E'.
    ENDTRY.

    "Get the job ID."
    DATA(lv_jobid) = lo_translationjob_result->get_jobid( ).

    "Wait for translate job to complete."
    DATA(lo_des_translation_result) = lo_xl8-
>describetexttranslationjob( iv_jobid = lv_jobid ).
    WHILE lo_des_translation_result->get_textxlationjobproperties( )-
>get_jobstatus( ) <> 'COMPLETED'.
      IF sy-index = 30.
```

```
        EXIT.                "Maximum 900 seconds."
      ENDIF.
      WAIT UP TO 30 SECONDS.
      lo_des_translation_result = lo_xl8->describetexttranslationjob( iv_jobid =
lv_jobid ).
    ENDWHILE.

    TRY.
        oo_result = lo_xl8->describetexttranslationjob(      "oo_result is
returned for testing purposes."
          iv_jobid        = lv_jobid ).
        MESSAGE 'Job description retrieved.' TYPE 'I'.
      CATCH /aws1/cx_xl8internalserverex.
        MESSAGE 'An internal server error occurred. Retry your request.' TYPE
'E'.
      CATCH /aws1/cx_xl8resourcenotfoundex.
        MESSAGE 'The resource you are looking for has not been found.' TYPE 'E'.
      CATCH /aws1/cx_xl8toomanyrequestsex.
        MESSAGE 'You have made too many requests within a short period of time.'
TYPE 'E'.
    ENDTRY.
```

- 有关 API 详细信息，请参阅《适用于 SAP ABAP 的AWS SDK 的 API 参考》中的以下主题。
  - DescribeTextTranslationJob
  - StartTextTranslationJob

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

标记您的资源

标签是您可以与 Amazon Translate 资源相关联的元数据。一个标签由一个键值对组成。您可以向并行数据和自定义术语资源添加标签。

标签有两个主要功能：整理资源以及提供基于标签的访问控制。您可以将标签添加到资源，然后创建 IAM 策略，以便根据资源的标签来允许或限制对资源的访问。

策略可以根据您的请求中提供的标签（请求标签）或与您正在调用的资源关联的标签（资源标签）来允许或禁止某项操作。有关通过 IAM 使用标签的更多信息，请参阅《IAM 用户指南》中的使用标签控制访问。

在 Amazon Translate 中使用标签的注意事项：

- 您最多可以为每个资源添加 50 个标签。
- 您可以在资源创建时或在资源创建之后的任何时候添加标签。
- 标签键是必填字段，但标签值是可选字段。
- 各个资源的标签不必是唯一的，但给定资源的标签必须具有唯一的键。
- 标签键和值区分大小写。
- 标签键的最大字符长度为 128 个字符，标签值的最大长度为 256 个字符。
- AWS 系统标签以标签键或值`aws:`中的前缀开头。您无法添加、编辑或删除带此前缀的标签名称或值。系统标签不包含在每个资源的标签限额中。

> **Note**
>
> 如果您计划在多个 AWS 服务和资源中使用标记架构，请记住，其他服务对允许的字符可能有不同的要求。

主题

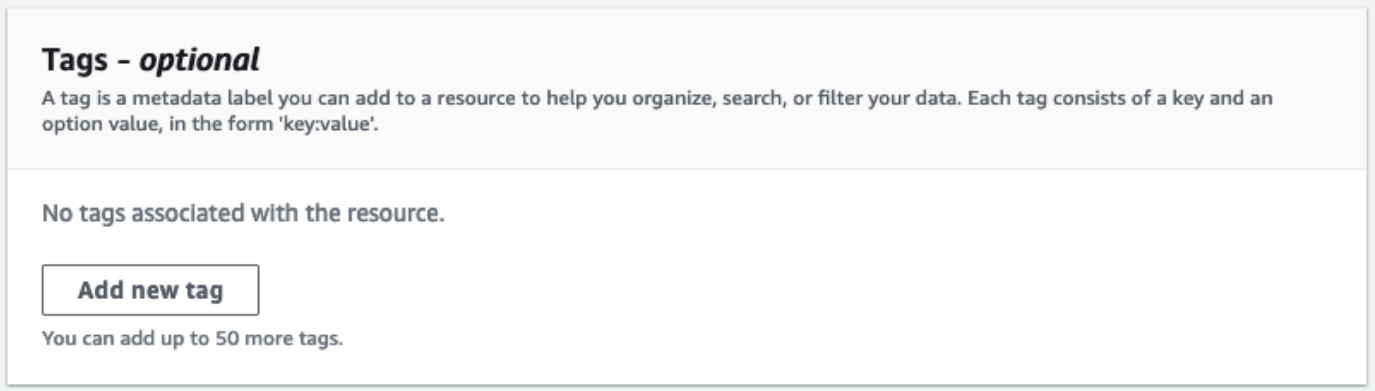
- 标记新资源
- 查看、更新和删除与资源关联的标签

标记新资源

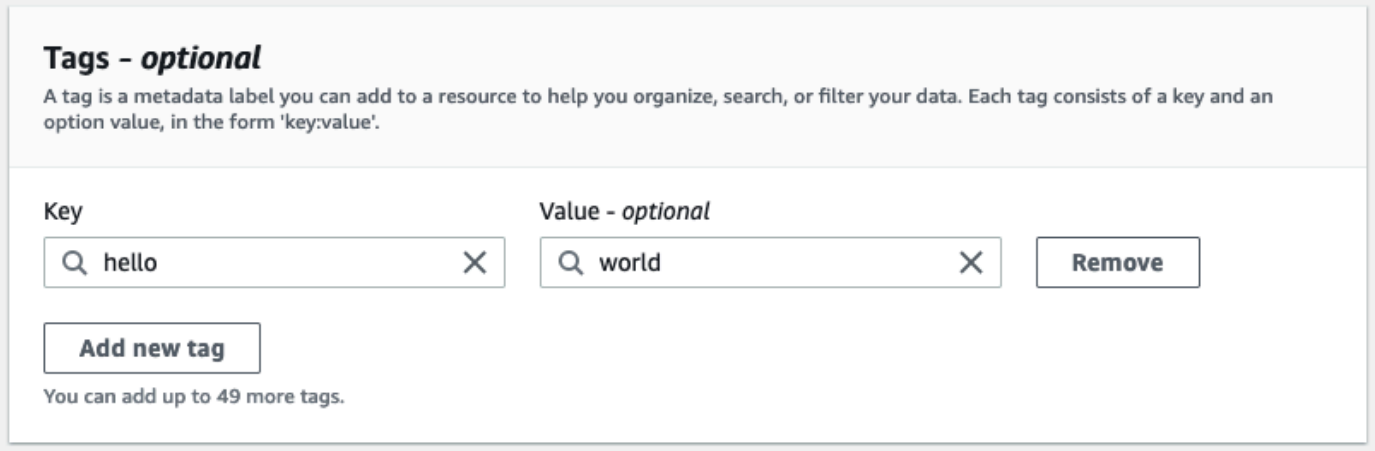
您可以在创建ParallelData或自定义术语资源时向其添加标签。

为资源添加标签（控制台）

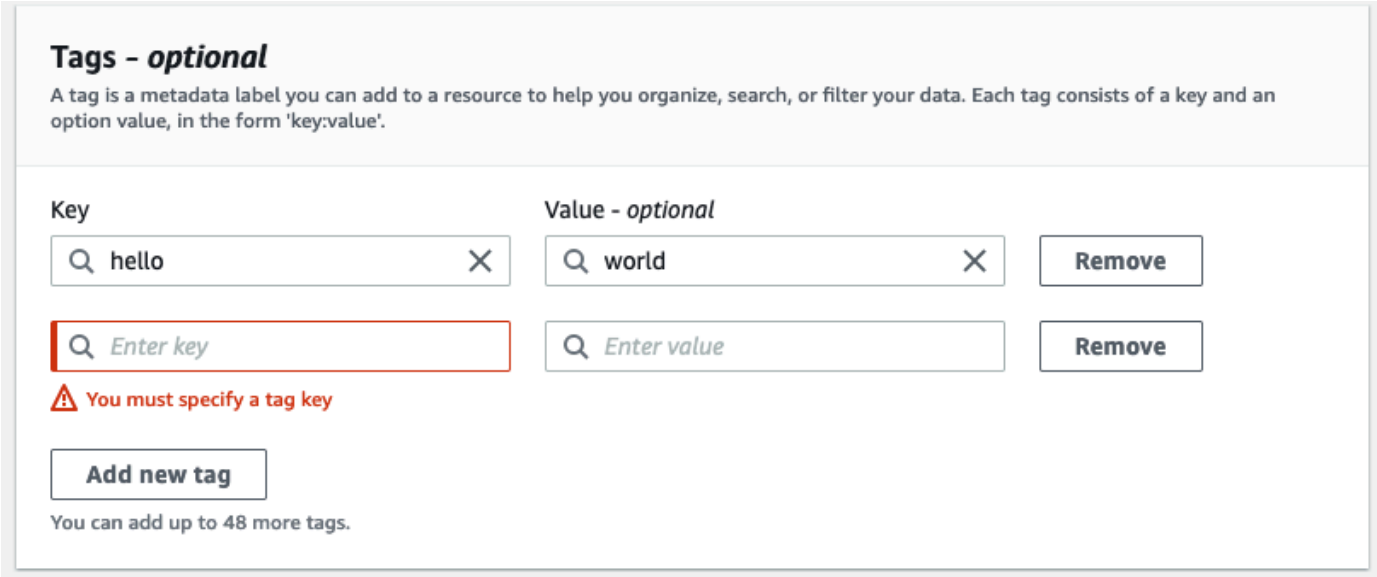
1. 登录 Amazon Translate 控制台。
2. 在左侧导航窗格中，选择要创建的资源（`Parallel data` 或 `Custom terminology`）。
3. 选择创建并行数据或创建术语。控制台会显示资源的主“创建”页面。在此页面的末尾，您将看到一个“标签 - 可选”面板。



4. 选择添加新标签，为资源添加标签。输入一个标签键和可选的标签值。



5. 重复步骤 4，直到您添加完所有标签。该资源的每个键必须是唯一的。



6. 选择创建并行数据或创建术语来创建资源。

您也可以使用 Amazon Translate CreateParallelDataAPI 操作添加标签。以下示例说明如何 create-parallel-data使用 CLI 命令添加标签。

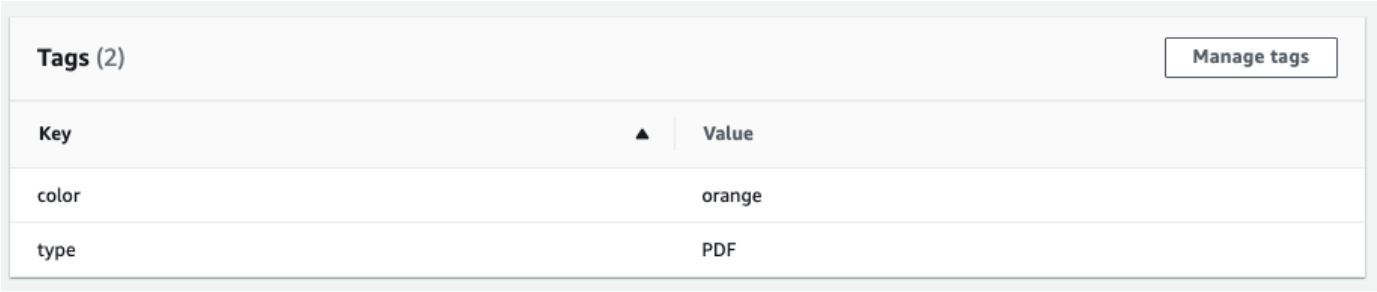
```
aws translate create-parallel-data \
--name "myTest" \
--parallel-data-config "{\"format\": \"CSV\",  \
             "S3Uri\": \"s3://test-input/TEST.csv\"}" \
--tags "[{\"Key\": \"color\",\"Value\": \"orange\"}]"
```

查看、更新和删除与资源关联的标签

您可以查看、更新或删除与 `Parallel data` 或 `Custom terminology` 资源关联的标签。

更新现有资源的标签（控制台）

1. 登录 Amazon Translate 控制台。
2. 从左侧导航窗格中选择 `Parallel data` 或 `Custom terminology`。
3. 选择包含要查看、更新或删除的标签的资源。控制台打开该资源的详细信息页面。
4. 向下滚动，直到看到标签面板。在这里，您可以看到与所选资源关联的所有标签。



选择管理标签以编辑或删除资源中的标签。

5. 选择要修改的文本，然后编辑标签。您也可以通过选择删除来删除标签。
6. 若要添加新标签，请选择添加新标签，然后在空白字段中输入键和值。
7. 修改完成后，选择保存。

Amazon Translate 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。

安全是双方共同承担 AWS 的责任。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 合规性计划的一部分，第三方审核人员将定期测试和验证安全性的有效性。要了解适用于 Amazon Translate 的合规性计划，请参阅合规性计划范围内的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。

此主题将帮助您了解如何在使用 AWS时应用责任共担模式。以下主题向您介绍如何进行配置 AWS 以满足您的安全和合规性目标。您还将学习如何使用其他 AWS 服务来帮助您监控和保护您的 AWS 资源。

主题

- Amazon Translate 中的数据保护
- 适用于 Amazon Translate 的 Identity and Access Management
- 监控 Amazon Translate
- Amazon Translate 的合规性验证
- Amazon Translate 中的故障恢复能力
- Amazon Translate 中的基础设施安全性
- Amazon Translate 和接口 VPC 端点 (AWS PrivateLink)

Amazon Translate 中的数据保护

Amazon Translate 符合 AWS 分担责任模式，其中包括数据保护的法规和指南。 AWS 负责保护运行所有 AWS 服务的全球基础架构。 AWS 保持对托管在此基础架构上的数据的控制，包括用于处理客户内容和个人数据的安全配置控制。 AWS 客户和 APN 合作伙伴，无论是作为数据控制者还是数据处理者，都应对他们在 AWS 云端存储的任何个人数据负责。

出于数据保护目的，我们建议您保护 AWS 账户凭证并使用 AWS Identity and Access Management (IAM) 设置角色，这样每位用户仅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）。
- 用于 SSL/TLS 与 AWS 资源通信。
- 使用设置 API 和用户活动日志 AWS CloudTrail。
- 使用 AWS 加密解决方案以及 AWS 服务中的所有默认安全控制。
- 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 Simple Storage Service (Amazon S3) 中的个人数据。

我们强烈建议您切勿将敏感的可识别信息（例如您客户的账号）放入自由格式字段（例如名称字段）。这包括您使用控制台、API 或使用 Amazon Translate 或其他 AWS 服务时 AWS SDKs。 AWS CLI您输入到 Amazon Translate 或其它服务中的任何数据都可能被选取以包含在诊断日志中。当您向外部服务器提供网址时，请勿在网址中包含凭证信息来验证您对该服务器的请求。

有关数据保护的更多信息，请参阅AWS 安全性博客 上的AWS 责任共担模式和 GDPR 博客文章。

主题

- 静态加密
- 传输中加密

静态加密

对于您使用 Amazon Translate 运行的批量翻译作业，您的翻译输入和输出都是静态加密的。但是，各自的加密方法都有所不同。

Amazon Translate 还使用使用默认密钥加密的 Amazon Elastic Block Store (Amazon EBS) 卷。

翻译输入

在使用 Amazon Translate 批量翻译文档时，您将一组输入文档存储在 Amazon S3 存储桶中。要静态加密这些文档，您可以使用 Amazon S3 提供的 SSE-S3 服务器端加密选项。使用此选项，每个对象均使用由 Amazon S3 管理的唯一密钥加密。

有关更多信息，请参阅《Amazon Simple Storage Service 用户指南》中的使用采用 Amazon S3 托管加密密钥的服务器端加密 (SSE-S3) 保护数据。

翻译输出

当 Amazon Translate 完成批量翻译任务时，它会将输出内容放入您 AWS 账户中的 Amazon S3 存储桶中。为了静态加密输出，Amazon Translate 使用 Amazon S3 提供的 SSE-KMS 加密选项。使用此选项，您的输出将使用存储在 AWS Key Management Service (AWS KMS) 中的密钥进行加密。

有关 SSE 的更多信息，请参阅《Amazon Simple Storage Service 用户指南》中的通过 AWS Key Management Service (SSE-KMS) 使用服务器端加密保护数据。

有关 KMS 密钥的更多信息，请参阅《AWS Key Management Service 开发人员指南》中的 AWS KMS keys。

Amazon Translate 可以使用以下任一类型的密钥：

AWS 托管式密钥

默认情况下，Amazon Translate 使用 AWS 托管式密钥。此类型的 KMS 密钥由您的账户创建 AWS 并存储在您的账户中。但是，您无法自己管理此 KMS 密钥，只能由 AWS代表您管理和使用。

客户托管密钥

或者，您可以选择使用客户托管密钥加密输出，该密钥是您在 AWS 账户中创建、拥有和管理的 KMS 密钥。

在使用您自己的 KMS 密钥之前，您必须为 Amazon Translate 用于访问 Amazon S3 中输出存储桶的 IAM 服务角色添加权限。如果您想使用其他 AWS 账户中的 KMS 密钥，还必须更新中的密钥策略 AWS KMS。有关更多信息，请参阅 自定义加密的必备权限。

在运行批量翻译作业时，您可以选择使用客户托管密钥。有关更多信息，请参阅 运行批量翻译作业。

传输中加密

为加密传输中数据，Amazon Translate 结合使用 TLS 1.2 和 AWS 证书。

适用于 Amazon Translate 的 Identity and Access Management

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以通过身份验证（登录）和授权（具有权限）使用 Amazon Translate 资源。您可以使用 IAM AWS 服务 ，无需支付额外费用。

主题

- 受众
- 使用身份进行身份验证
- 使用策略管理访问
- Amazon Translate 如何与 IAM 配合使用
- Amazon Translate 基于身份的策略示例
- AWS Amazon Translate 的托管政策
- Amazon Translate 身份和访问问题排查

受众

您的使用方式 AWS Identity and Access Management (IAM) 会有所不同，具体取决于您在 Amazon Translate 中所做的工作。

服务用户 - 如果您使用 Amazon Translate 服务来完成工作，您的管理员会为您提供所需的凭证和权限。随着您使用更多 Amazon Translate 特征来完成工作，您可能需要额外权限。了解如何管理访问权限有助于您向管理员请求适合的权限。如果您无法访问 Amazon Translate 中的特征，请参阅 Amazon Translate 身份和访问问题排查。

服务管理员 – 如果您在公司负责管理 Amazon Translate 资源，您可能对 Amazon Translate 具有完全访问权限。您有责任确定您的服务用户应访问哪些 Amazon Translate 特征和资源。然后，您必须向 IAM 管理员提交请求以更改服务用户的权限。请查看该页面上的信息以了解 IAM 的基本概念。要了解有关您的公司如何将 IAM 与 Amazon Translate 搭配使用的更多信息，请参阅 Amazon Translate 如何与 IAM 配合使用。

IAM 管理员 – 如果您是 IAM 管理员，您可能需要了解如何编写策略以管理对 Amazon Translate 的访问的详细信息。要查看您可在 IAM 中使用的 Amazon Translate 基于身份的策略示例，请参阅 Amazon Translate 基于身份的策略示例。

使用身份进行身份验证

身份验证是您 AWS 使用身份凭证登录的方式。您必须以 IAM 用户身份或通过担 AWS 账户根用户任 IAM 角色进行身份验证（登录 AWS）。

您可以使用通过身份源提供的凭据以 AWS 联合身份登录。 AWS IAM Identity Center （IAM Identity Center）用户、贵公司的单点登录身份验证以及您的 Google 或 Facebook 凭据就是联合身份的示例。

当您以联合身份登录时，您的管理员以前使用 IAM 角色设置了身份联合验证。当你使用联合访问 AWS 时，你就是在间接扮演一个角色。

根据您的用户类型，您可以登录 AWS Management Console 或 AWS 访问门户。有关登录的更多信息 AWS，请参阅《AWS 登录 用户指南》中的如何登录到您 AWS 账户的。

如果您 AWS 以编程方式访问，则会 AWS 提供软件开发套件 (SDK) 和命令行接口 (CLI)，以便使用您的凭据对请求进行加密签名。如果您不使用 AWS 工具，则必须自己签署请求。有关使用推荐的方法自行签署请求的更多信息，请参阅《IAM 用户指南》中的用于签署 API 请求的AWS 签名版本 4。

无论使用何种身份验证方法，您都可能需要提供其他安全信息。例如， AWS 建议您使用多重身份验证 (MFA) 来提高账户的安全性。要了解更多信息，请参阅《AWS IAM Identity Center 用户指南》中的多重身份验证和《IAM 用户指南》中的 IAM 中的AWS 多重身份验证。

AWS 账户 root 用户

创建时 AWS 账户，首先要有一个登录身份，该身份可以完全访问账户中的所有资源 AWS 服务 和资源。此身份被称为 AWS 账户 root 用户，使用您创建账户时使用的电子邮件地址和密码登录即可访问该身份。强烈建议您不要使用根用户执行日常任务。保护好根用户凭证，并使用这些凭证来执行仅根用户可以执行的任务。有关要求您以根用户身份登录的任务的完整列表，请参阅 IAM 用户指南中的需要根用户凭证的任务。

联合身份

作为最佳实践，要求人类用户（包括需要管理员访问权限的用户）使用与身份提供商的联合身份验证 AWS 服务 通过临时证书进行访问。

联合身份是指您的企业用户目录、Web 身份提供商、Identity C enter 目录中的用户，或者任何使用 AWS 服务 通过身份源提供的凭据进行访问的用户。 AWS Directory Service当联合身份访问时 AWS 账户，他们将扮演角色，角色提供临时证书。

要集中管理访问权限，建议您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中创建用户和群组，也可以连接并同步到您自己的身份源中的一组用户和群组，以便在您的所有 AWS 账户 和应用程序中使用。有关 IAM Identity Center 的信息，请参阅 AWS IAM Identity Center 用户指南中的什么是 IAM Identity Center？。

IAM 用户和群组

I AM 用户是您 AWS 账户 内部对个人或应用程序具有特定权限的身份。在可能的情况下，我们建议使用临时凭证，而不是创建具有长期凭证（如密码和访问密钥）的 IAM 用户。但是，如果您有一些特定

的使用场景需要长期凭证以及 IAM 用户，建议您轮换访问密钥。有关更多信息，请参阅《IAM 用户指南》中的对于需要长期凭证的用例，应在需要时更新访问密钥。

IAM 组是一个指定一组 IAM 用户的身份。您不能使用组的身份登录。您可以使用组来一次性为多个用户指定权限。如果有大量用户，使用组可以更轻松地管理用户权限。例如，您可以拥有一个名为的群组，IAMAdmins并向该群组授予管理 IAM 资源的权限。

用户与角色不同。用户唯一地与某个人员或应用程序关联，而角色旨在让需要它的任何人代入。用户具有永久的长期凭证，而角色提供临时凭证。要了解更多信息，请参阅《IAM 用户指南》中的 IAM 用户的使用案例。

IAM 角色

I AM 角色是您内部具有特定权限 AWS 账户 的身份。它类似于 IAM 用户，但与特定人员不关联。要在中临时担任 IAM 角色 AWS Management Console，您可以从用户切换到 IAM 角色（控制台）。您可以通过调用 AWS CLI 或 AWS API 操作或使用自定义 URL 来代入角色。有关使用角色的方法的更多信息，请参阅《IAM 用户指南》中的代入角色的方法。

具有临时凭证的 IAM 角色在以下情况下很有用：

- 联合用户访问：要向联合身份分配权限，请创建角色并为角色定义权限。当联合身份进行身份验证时，该身份将与角色相关联并被授予由此角色定义的权限。有关用于联合身份验证的角色的信息，请参阅《IAM 用户指南》中的针对第三方身份提供商创建角色（联合身份验证）。如果您使用 IAM Identity Center，则需要配置权限集。为控制您的身份在进行身份验证后可以访问的内容，IAM Identity Center 将权限集与 IAM 中的角色相关联。有关权限集的信息，请参阅《AWS IAM Identity Center 用户指南》中的权限集。
- 临时 IAM 用户权限：IAM 用户可代入 IAM 用户或角色，以暂时获得针对特定任务的不同权限。
- 跨账户存取：您可以使用 IAM 角色以允许不同账户中的某个人（可信主体）访问您的账户中的资源。角色是授予跨账户访问权限的主要方式。但是，对于某些资源 AWS 服务，您可以将策略直接附加到资源（而不是使用角色作为代理）。要了解用于跨账户访问的角色和基于资源的策略之间的差别，请参阅 IAM 用户指南中的 IAM 中的跨账户资源访问。
- 跨服务访问 — 有些 AWS 服务 使用其他 AWS 服务服务中的功能。例如，当您在服务中拨打电话时，该服务通常会在 Amazon 中运行应用程序 EC2 或在 Amazon S3 中存储对象。服务可能会使用发出调用的主体的权限、使用服务角色或使用服务相关角色来执行此操作。
  - 转发访问会话 (FAS) — 当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他

AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两项操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。

- 服务角色 - 服务角色是服务代表您在您的账户中执行操作而分派的 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。
- 服务相关角色-服务相关角色是一种与服务相关联的服务角色。 AWS 服务服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

- 在 A@@ mazon 上运行的应用程序 EC2 — 您可以使用 IAM 角色管理在 EC2 实例上运行并发出 AWS CLI 或 AWS API 请求的应用程序的临时证书。这比在 EC2 实例中存储访问密钥更可取。要为 EC2 实例分配 AWS 角色并使其可供其所有应用程序使用，您需要创建一个附加到该实例的实例配置文件。实例配置文件包含该角色，并允许在 EC2 实例上运行的程序获得临时证书。有关更多信息，请参阅 IAM 用户指南中的使用 IAM 角色向在 A mazon EC2 实例上运行的应用程序授予权限。

使用策略管理访问

您可以 AWS 通过创建策略并将其附加到 AWS 身份或资源来控制中的访问权限。策略是其中的一个对象 AWS ，当与身份或资源关联时，它会定义其权限。 AWS 在委托人（用户、root 用户或角色会话）发出请求时评估这些策略。策略中的权限确定是允许还是拒绝请求。大多数策略都以 JSON 文档的 AWS 形式存储在中。有关 JSON 策略文档的结构和内容的更多信息，请参阅 IAM 用户指南中的 JSON 策略概览。

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

默认情况下，用户和角色没有权限。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

IAM 策略定义操作的权限，无关乎您使用哪种方法执行操作。例如，假设您有一个允许 `iam:GetRole` 操作的策略。拥有该策略的用户可以从 AWS Management Console AWS CLI、或 AWS API 获取角色信息。

基于身份的策略

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户托管策略定义自定义 IAM 权限。

基于身份的策略可以进一步归类为内联策略或托管式策略。内联策略直接嵌入单个用户、组或角色中。托管策略是独立的策略，您可以将其附加到中的多个用户、群组和角色 AWS 账户。托管策略包括 AWS 托管策略和客户托管策略。要了解如何在托管策略和内联策略之间进行选择，请参阅《IAM 用户指南》中的在托管策略与内联策略之间进行选择。

基于资源的策略

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

基于资源的策略是位于该服务中的内联策略。您不能在基于资源的策略中使用 IAM 中的 AWS 托管策略。

访问控制列表 (ACLs)

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

Amazon S3 和 Amazon VPC 就是支持的服务示例 ACLs。 AWS WAF要了解更多信息 ACLs，请参阅《亚马逊简单存储服务开发者指南》中的访问控制列表 (ACL) 概述。

其他策略类型

AWS 支持其他不太常见的策略类型。这些策略类型可以设置更常用的策略类型向您授予的最大权限。

- 权限边界：权限边界是一个高级特征，用于设置基于身份的策略可以为 IAM 实体（IAM 用户或角色）授予的最大权限。您可为实体设置权限边界。这些结果权限是实体基于身份的策略及其权限边界的交集。在 `Principal` 中指定用户或角色的基于资源的策略不受权限边界限制。任一项策略中的显式拒绝将覆盖允许。有关权限边界的更多信息，请参阅IAM 用户指南中的 IAM 实体的权限边界。
- 服务控制策略 (SCPs)- SCPs 是指定组织或组织单位 (OU) 的最大权限的 JSON 策略 AWS Organizations。 AWS Organizations 是一项用于对您的企业拥有的多 AWS 账户 项进行分组和集中管理的服务。如果您启用组织中的所有功能，则可以将服务控制策略 (SCPs) 应用于您的任何或所有帐户。SCP 限制成员账户中的实体（包括每个 AWS 账户根用户实体）的权限。有关 Organization SCPs s 和的更多信息，请参阅《AWS Organizations 用户指南》中的服务控制策略。
- 资源控制策略 (RCPs) — RCPs 是 JSON 策略，您可以使用它来设置账户中资源的最大可用权限，而无需更新附加到您拥有的每个资源的 IAM 策略。RCP 限制成员账户中资源的权限，并

可能影响身份（包括身份）的有效权限 AWS 账户根用户，无论这些身份是否属于您的组织。有关 Organizations 的更多信息 RCPs，包括 AWS 服务 该支持的列表 RCPs，请参阅《AWS Organizations 用户指南》中的资源控制策略 (RCPs)。

- 会话策略：会话策略是当您以编程方式为角色或联合用户创建临时会话时作为参数传递的高级策略。结果会话的权限是用户或角色的基于身份的策略和会话策略的交集。权限也可以来自基于资源的策略。任一项策略中的显式拒绝将覆盖允许。有关更多信息，请参阅 IAM 用户指南中的会话策略。

多个策略类型

当多个类型的策略应用于一个请求时，生成的权限更加复杂和难以理解。要了解在涉及多种策略类型时如何 AWS 确定是否允许请求，请参阅 IAM 用户指南中的策略评估逻辑。

Amazon Translate 如何与 IAM 配合使用

在使用 IAM 管理对 Amazon Translate 的访问权限之前，您应该了解哪些 IAM 特征可用于 Amazon Translate。

可以与 Amazon Translate 配合使用的 IAM 特征

IAM 特征	Amazon Translate 支持
基于身份的策略	是
基于资源的策略	否
策略操作	是
策略资源	是
策略条件键（特定于服务）	是
ACLs	否
ABAC（策略中的标签）	部分
临时凭证	是
转发访问会话（FAS）	是

IAM 特征	Amazon Translate 支持
服务角色	是
服务相关角色	否

要全面了解 Amazon Translate 和其他 AWS 服务如何与大多数 IAM 功能配合使用，请参阅 IAM 用户指南中与 IAM 配合使用的AWS 服务。

Amazon Translate 基于身份的策略

支持基于身份的策略：是

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户管理型策略定义自定义 IAM 权限。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素引用。

Amazon Translate 基于身份的策略示例

要查看 Amazon Translate 基于身份的策略的示例，请参阅 Amazon Translate 基于身份的策略示例。

Amazon Translate 基于资源的策略

支持基于资源的策略：否

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

要启用跨账户访问，您可以将整个账户或其他账户中的 IAM 实体指定为基于资源的策略中的主体。将跨账户主体添加到基于资源的策略只是建立信任关系工作的一半而已。当委托人和资源处于不同位置时 AWS 账户，可信账户中的 IAM 管理员还必须向委托人实体（用户或角色）授予访问资源的权限。他们通过将基于身份的策略附加到实体以授予权限。但是，如果基于资源的策略向同一个账户中的主体授予

访问权限，则不需要额外的基于身份的策略。有关更多信息，请参阅《IAM 用户指南》中的 IAM 中的跨账户资源访问。

Amazon Translate 的策略操作

支持策略操作：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

有关 Amazon Translate 操作的列表，请参阅《服务授权参考》中的 Amazon Translate 定义的操作。

Amazon Translate 中的策略操作在操作前使用以下前缀：

```
translate
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。

```
"Action": [
      "translate:ListLanguages",
      "translate:TranslateText"
         ]
```

您也可以使用通配符（*）指定多个操作。例如，要指定以单词 `List` 开头的所有操作，包括以下操作：

```
"Action": "translate:List*"
```

请勿使用通配符来指定一个服务的所有操作。在确定策略中的权限时，使用授予最低权限的最佳实操。

要查看 Amazon Translate 基于身份的策略的示例，请参阅 Amazon Translate 基于身份的策略示例。

Amazon Translate 的策略资源

支持策略资源：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

`Resource` JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 `Resource` 或 `NotResource` 元素。作为最佳实践，请使用其 Amazon 资源名称（ARN）指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符（*）指示语句应用于所有资源。

```
"Resource": "*"
```

要查看 Amazon Translate 资源类型及其列表 ARNs，请参阅服务授权参考中的 Amazon Translate 定义的资源。要了解您可以使用哪些操作指定每个资源的 ARN，请参阅 Amazon Translate 定义的操作。

有关如何使用 Amazon Translate 策略中资源的示例，请参阅 在策略中指定资源。

Amazon Translate 的策略条件键

支持特定于服务的策略条件键：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 `Condition` 元素（或 `Condition` 块）中，可以指定语句生效的条件。`Condition` 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 `Condition` 元素，或在单个 `Condition` 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑OR运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM 策略元素：变量和标签。

AWS 支持全局条件密钥和特定于服务的条件密钥。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的AWS 全局条件上下文密钥。

要查看 Amazon Translate 条件键的列表，请参阅《服务授权参考》中的 Amazon Translate 的条件键。要了解您可以对哪些操作和资源使用条件键，请参阅 Amazon Translate 定义的操作。

要查看 Amazon Translate 基于身份的策略的示例，请参阅 Amazon Translate 基于身份的策略示例。

ACLs 在亚马逊 Translate 中

支持 ACLs：否

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

ABAC 与 Amazon Translate

支持 ABAC（策略中的标签）：部分支持

基于属性的访问控制（ABAC）是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以将标签附加到 IAM 实体（用户或角色）和许多 AWS 资源。标记实体和资源是 ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，您需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的条件元素中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件键，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件键，则该值为部分。

有关 ABAC 的更多信息，请参阅《IAM 用户指南》中的使用 ABAC 授权定义权限。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的使用基于属性的访问权限控制（ABAC）。

有关标记 Amazon Translate 资源的更多信息，请参阅 标记您的资源。

将临时凭证用于 Amazon Translate

支持临时凭证：是

当你使用临时证书登录时，有些 AWS 服务 不起作用。有关更多信息，包括哪些 AWS 服务 适用于临时证书，请参阅 IAM 用户指南中的AWS 服务 与 IAM 配合使用的信息。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以用户身份登录控制台，然后切换角色时，您还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的从用户切换到 IAM 角色（控制台）。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。 AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 IAM 中的临时安全凭证。

Amazon Translate 的转发访问会话

支持转发访问会话（FAS）：是

当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两项操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。

Amazon Translate 的服务角色

支持服务角色：是

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。

> **Warning**
>
> 更改服务角色的权限可能会破坏 Amazon Translate 功能。仅当 Amazon Translate 提供相关指导时才编辑服务角色。

要使用 Amazon Translate 异步操作，您必须授予 Amazon Translate 访问包含您的输入文档的 Amazon S3 存储桶的访问权限。为此，您可以在自己的账户中创建一个服务角色，并使用信任策略来信任 Amazon Translate 服务主体。

有关策略示例，请参阅 批量翻译作业的先决条件。

Amazon Translate 的服务相关角色

支持服务相关角色：否

服务相关角色是一种与服务相关联的 AWS 服务服务角色。服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

有关创建或管理服务相关角色的详细信息，请参阅能够与 IAM 搭配使用的AWS 服务。在表中查找服务相关角色列中包含 Yes 的表。选择是链接以查看该服务的服务相关角色文档。

Amazon Translate 基于身份的策略示例

原定设置情况下，用户和角色没有创建或修改 Amazon Translate 资源的权限。他们也无法使用 AWS Management Console AWS CLI、或 AWS API 执行任务。IAM 管理员必须创建 IAM policy，以便授予权限以对所需的特定资源执行特定的 API 操作。然后，管理员必须将这些策略附加到需要这些权限的用户或角色。

要了解如何使用以下示例 JSON 策略文档创建 IAM 基于身份的策略，请参阅《IAM 用户指南》中的 在 JSON 选项卡上创建策略。

主题

- 基于身份的策略最佳实操
- 允许访问 Amazon Translate 控制台
- 允许用户查看他们自己的权限
- 在策略中指定资源
- 将客户托管密钥与自定义术语结合使用的权限

基于身份的策略最佳实操

基于身份的策略确定某个人是否可以创建、访问或删除您账户中的 Amazon Translate 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管式策略或工作职能的AWS 托管式策略。
- 应用最低权限：在使用 IAM 策略设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。

- 使用 IAM 策略中的条件进一步限制访问权限：您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定的方式使用的，则也可以使用条件来授予对服务操作的访问权限 AWS 服务，例如 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件。
- 使用 IAM Access Analyzer 验证您的 IAM 策略，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM 策略语言（JSON）和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM Access Analyzer 验证策略。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的使用 MFA 保护 API 访问。

有关 IAM 中的最佳实操的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

允许访问 Amazon Translate 控制台

要访问 Amazon Translate 控制台，您必须具有一组最低的权限。这些权限必须允许您列出和查看有关您 AWS 账户中的 Amazon Translate 资源的详细信息。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户、组或角色），控制台将无法按预期正常运行。

要获得 Amazon Translate 控制台权限，您可以将`TranslateFullAccess` AWS 托管策略附加到实体。有关更多信息，请参阅 AWS Amazon Translate 的托管政策。

您还需要以下策略所示操作的权限。这些权限包含在 `TranslateFullAccess` 策略中。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iam:ListRoles",
                "iam:GetRole",
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation"
             ],
            "Resource": "*"

        }
```

```
    ]
}
```

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。有关更多信息，请参阅《IAM 用户指南》中的为用户添加权限。

允许用户查看他们自己的权限

该示例说明了您如何创建策略，以允许 IAM 用户查看附加到其用户身份的内联和托管式策略。此策略包括在控制台上或使用 AWS CLI 或 AWS API 以编程方式完成此操作的权限。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

在策略中指定资源

对于许多 Amazon Translate API 操作，您可以通过指定允许（或不允许）执行该操作的资源来限制策略的范围。有关可以指定资源的操作列表，请参阅 Amazon Translate 定义的操作。您可以在策略中指定以下资源：

- Custom terminology – 采用以下 ARN 格式：

  arn:*partition*:translate:*region*:*account*:terminology/*terminology-name*/LATEST

- Parallel data – 采用以下 ARN 格式：

  arn:*partition*:translate:*region*:*account*:parallel-data/*parallel-data-name*

您可以使用通配符指定策略中的多个资源。以下示例策略允许所有自定义术语资源用于所有 Amazon Translate 操作。

Example

```
{
        "Sid": "Example1",
        "Effect": "Allow",
        "Action": "translate:*",
        "Resource": [
             "arn:aws:translate:us-west-2:123456789012:terminology/*"
        ]
}
```

以下示例策略拒绝访问 GetParallelData 操作的特定并行数据资源。

Example

```
{
        "Sid": "Example2",
        "Effect": "Deny",
        "Action": "translate:GetParallelData",
        "Resource": [
             "arn:aws:translate:us-west-2:123456789012:parallel-data/test-parallel-
data"
        ]
}
```

将客户托管密钥与自定义术语结合使用的权限

如果您将 AWS Key Management Service (AWS KMS) 客户托管密钥与 Amazon Translate 自定义术语一起使用，则可能需要在 KMS 密钥策略中获得额外的权限。

要使用客户托管密钥调用 `ImportTerminology` 操作，请将以下权限添加到现有 KMS 密钥政策。

```
{
    "Id": "key-consolepolicy-3",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow access for use with Amazon Translate",
            "Effect": "Allow",
            "Principal": {
                "AWS": "IAM USER OR ROLE ARN"
            },
            "Action": [
                "kms:CreateAlias",
                "kms:CreateGrant",
                "kms:DescribeKey",
                "kms:GenerateDataKey",
                "kms:GetKeyPolicy",
                "kms:PutKeyPolicy",
                "kms:RetireGrant"
            ],
            "Resource": "*"
        }
    ]
}
```

要使用 KMS 客户托管密钥为已导入的自定义术语调用 `GetTerminology` 操作，请在 KMS 密钥政策中添加以下权限。

```
{
    "Id": "key-consolepolicy-3",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow access for use with Amazon Translate",
            "Effect": "Allow",
            "Principal": {
```

```
                "AWS": "IAM USER OR ROLE ARN"
            },
            "Action": [
                "kms:Decrypt",
                "kms:GetKeyPolicy",
                "kms:PutKeyPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

要对使用客户托管密钥导入的自定义术语调用`ListTerminologies`或`DeleteTermionlogy`操作，您无需拥有任何特殊 AWS KMS 权限。

要将客户托管密钥与所有自定义术语操作结合使用，请在 KMS 密钥政策中添加以下权限。

```
{
    "Id": "key-consolepolicy-3",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow access for use with Amazon Translate",
            "Effect": "Allow",
            "Principal": {
                "AWS": "IAM USER OR ROLE ARN"
            },
            "Action": [
                "kms:CreateGrant",
                "kms:Decrypt",
                "kms:DescribeKey",
                "kms:GenerateDataKey",
                "kms:GetKeyPolicy",
                "kms:PutKeyPolicy",
                "kms:RetireGrant"
            ],
            "Resource": "*"
        }
    ]
}
```

有关 Amazon Translate 操作和资源的详细信息，请参阅《服务授权参考》中的 Amazon Translate 的操作、资源和条件密钥。

AWS Amazon Translate 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： TranslateFullAccess
- AWS 托管策略： TranslateReadOnly
- Amazon Translate 更新 AWS 了托管政策

AWS 托管策略： TranslateFullAccess

该政策授予对 Amazon Translate 资源、Amazon Comprehend DetectDominantLanguage API 操作和所需的 API 操作的完全访问权限。 CloudWatch 该策略还授予 Amazon S3 存储桶和 IAM 角色的列出和获取权限。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "translate:*",
                "comprehend:DetectDominantLanguage",
                "cloudwatch:GetMetricStatistics",
                "cloudwatch:ListMetrics",
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation",
                "iam:ListRoles",
                "iam:GetRole"
            ],
```

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

AWS 托管策略： TranslateReadOnly

该策略授予访问不修改与您账户关联的资源的 Amazon Translate API 操作的权限。该政策还授予访问亚马逊 Comprehend DetectDominantLanguage API 操作和所需的 API 操作的权限。 CloudWatch

```
{
  "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "translate:TranslateText",
                "translate:TranslateDocument",
                "translate:GetTerminology",
                "translate:ListTerminologies",
                "translate:ListTextTranslationJobs",
                "translate:DescribeTextTranslationJob",
                "translate:GetParallelData",
                "translate:ListParallelData",
                "comprehend:DetectDominantLanguage",
                "cloudwatch:GetMetricStatistics",
                "cloudwatch:ListMetrics"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Amazon Translate 更新 AWS 了托管政策

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

更改	描述	日期
TranslateReadOnly – 对现有策略的更新	Amazon Translate 现在允许 TranslateReadOnly 政策中的`TranslateDocument` 操作	2023 年 5 月 23 日
Amazon Translate 已开始跟踪更改	Amazon Translate 开始跟踪其 AWS 托管政策的变更。	2023 年 5 月 23 日

Amazon Translate 身份和访问问题排查

您可以使用以下信息，帮助诊断和修复在使用 Amazon Translate 和 IAM 时可能遇到的常见问题。

主题

- 我无权在 Amazon Translate 中执行操作
- 我无权执行 iam：PassRole
- 我想允许我以外的人访问我的 Amazon Translate 资源 AWS 账户

我无权在 Amazon Translate 中执行操作

如果您收到错误提示，指明您无权执行某个操作，则必须更新策略以允许执行该操作。

当 `mateojackson` IAM 用户尝试使用控制台查看有关虚构 *`my-example-widget`* 资源的详细信息，但不拥有虚构 `translate:`*`GetWidget`* 权限时，会发生以下示例错误。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 translate:GetWidget on resource: my-example-widget
```

在此情况下，Mateo 的策略必须更新以允许其使用 `translate:`*`GetWidget`* 操作访问 *`my-example-widget`* 资源。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我无权执行 iam：PassRole

如果您收到一个错误，指明您无权执行 `iam:PassRole` 操作，则必须更新策略以允许您将角色传递给 Amazon Translate。

有些 AWS 服务 允许您将现有角色传递给该服务，而不是创建新的服务角色或服务相关角色。为此，您必须具有将角色传递到服务的权限。

当名为 `marymajor` 的 IAM 用户尝试使用控制台在 Amazon Translate 中执行操作时，会发生以下示例错误。但是，服务必须具有服务角色所授予的权限才可执行此操作。Mary 不具有将角色传递到服务的权限。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在这种情况下，必须更新 Mary 的策略以允许她执行 `iam:PassRole` 操作。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我想允许我以外的人访问我的 Amazon Translate 资源 AWS 账户

您可以创建一个角色，以便其他账户中的用户或您组织外的人员可以使用该角色来访问您的资源。您可以指定谁值得信赖，可以代入角色。对于支持基于资源的策略或访问控制列表 (ACLs) 的服务，您可以使用这些策略向人们授予访问您的资源的权限。

要了解更多信息，请参阅以下内容：

- 要了解 Amazon Translate 是否支持这些特征，请参阅 Amazon Translate 如何与 IAM 配合使用。
- 要了解如何提供对您拥有的资源的访问权限 AWS 账户 ，请参阅 IAM 用户指南中的向您拥有 AWS 账户 的另一个 IAM 用户提供访问权限。
- 要了解如何向第三方提供对您的资源的访问权限 AWS 账户，请参阅 IAM 用户指南中的向第三方提供访问权限。 AWS 账户
- 要了解如何通过身份联合验证提供访问权限，请参阅《IAM 用户指南》中的为经过外部身份验证的用户（身份联合验证）提供访问权限。
- 要了解使用角色和基于资源的策略进行跨账户访问之间的差别，请参阅《IAM 用户指南》中的 IAM 中的跨账户资源访问。

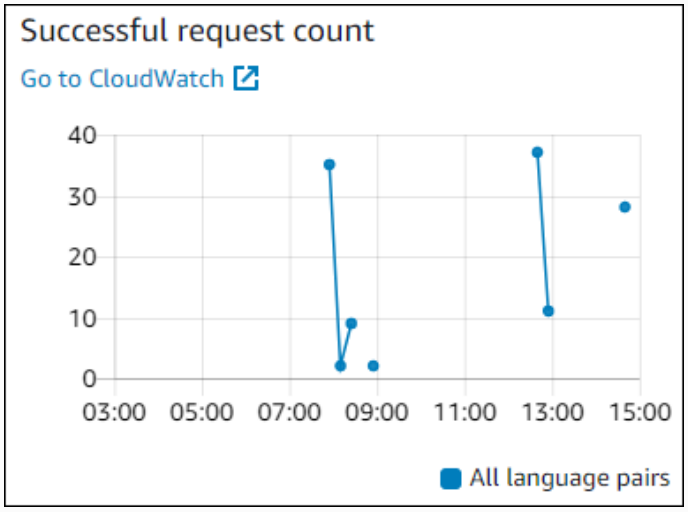
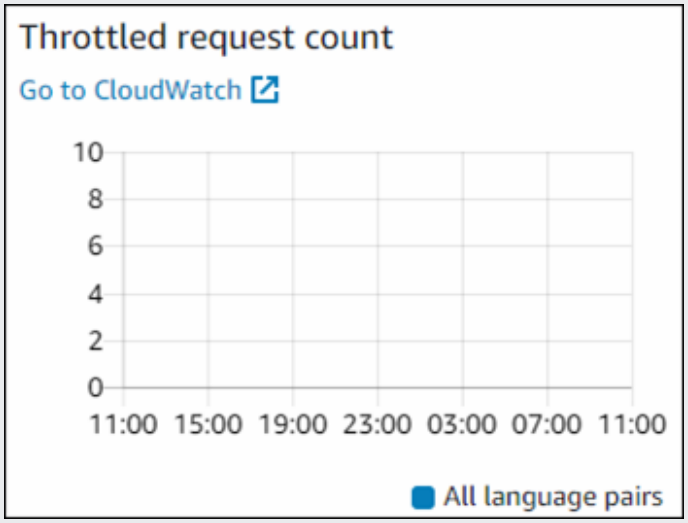
监控 Amazon Translate

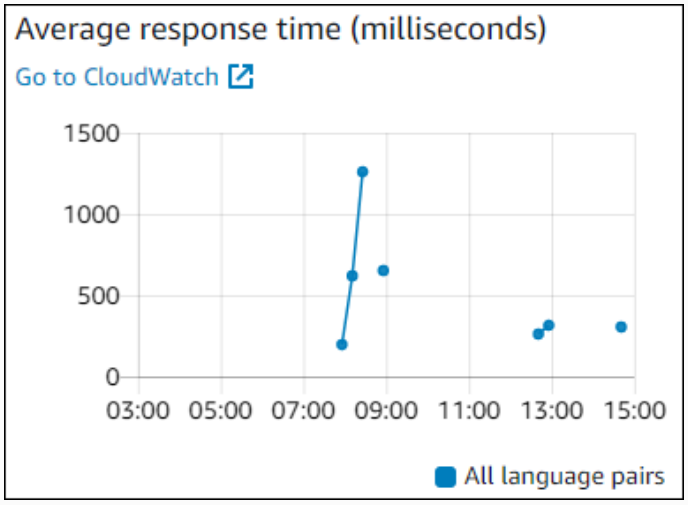
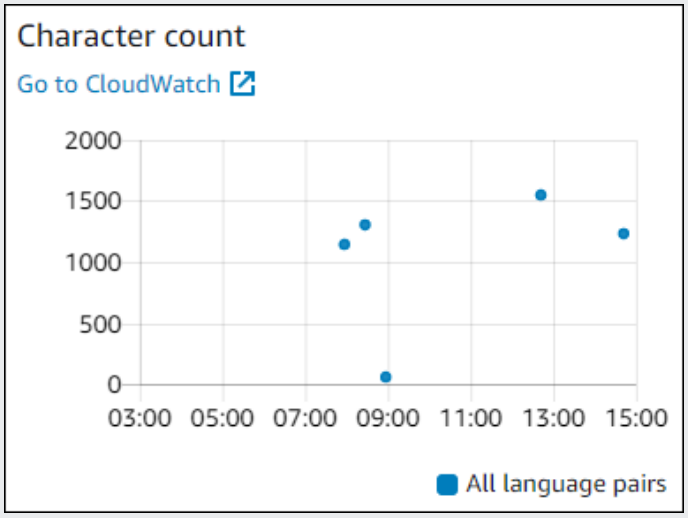
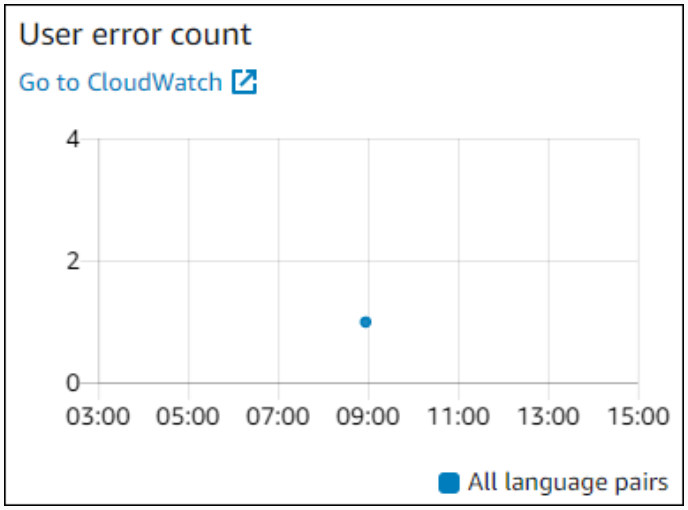
监控是维护 Amazon Translate 和您的解决方案的可靠性、可用性和性能的重要组成部分。 AWS 提供了可用于监控 Amazon Translate 的各种工具。您可以配置其中的一些工具来为您监控解决方案。建议您尽可能实现监控任务自动化。

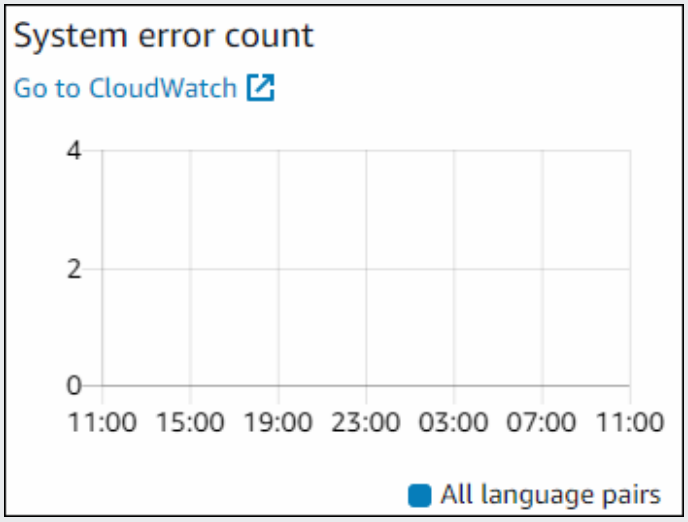
Amazon Translate 提供了预配置图表，可向您展示对您的解决方案最重要的指标。每个图表提供一个了解您的解决方案性能的窗口。要获取您的解决方案随着时间的推移如何表现的不同视角，您可以更改这些图表显示的时间范围。

您也可以使用亚马逊 CloudWatch 来监控 Amazon Translate。借 CloudWatch助，您可以自动监控解决方案的特定指标。只要某个指标超出您设定的阈值，您就会收到一条通知。您还可以使用 CloudWatch API 创建适合您需求的自定义监控应用程序。有关更多信息，请参阅《亚马逊 CloudWatch 用户指南》 CloudWatch中的 “什么是亚马逊”。

下表介绍了 Amazon Translate 提供的每个预配置图表。

图表	描述
	成功请求计数 指定时间段内向 Amazon Translate 发出的成功请求数。
	受限请求计数 指定时间段内 Amazon Translate 收到的受限请求数。使用此信息确定您的应用程序是否太快向 Amazon Translate 发送请求。

图表	描述
	平均响应时间 指定时间段内 Amazon Translate 处理您的请求所花费的平均时长。
	字符计数 指定时间段内您发送给 Amazon Translate 的字符总数。这是您要支付的字符数。
	用户错误计数 指定时间段内发生的用户错误数。用户错误位于 HTTP 错误代码范围 400-499。

图表	描述
	系统错误计数 指定时间段内发生的系统错误数。系统错误位于 HTTP 错误代码范围 500-599。

监控 Amazon Translate

通过亚马逊 CloudWatch，您可以获取各个亚马逊翻译业务的指标或账户的全球亚马逊翻译指标。使用指标跟踪 Amazon Translate 解决方案的运行状况并设置警报，以便在一个或多个指标超出定义的阈值时通知您。例如，您可以监控在特定的时间段内发送给 Amazon Translate 的请求数量、查看请求的延迟，或在错误数超出阈值时发出警报。

了解 Amazon T CloudWatch ranslate 的指标

要获取 Amazon Translate 操作的指标，请指定以下信息：

- 指标维度。维度 是用来标识指标的一组名称/值对。Amazon Translate 有两个维度：
  - `Operation`
  - `Language pair`
- 指标名称，如 `SuccessfulRequestCount` 或 `RequestCharacters`。要获得指标的完整列表，请参阅 CloudWatch Amazon Translate 的指标。

您可以通过 AWS Management Console、或 CloudWatch API 获取 Amazon Trans AWS CLI late 的指标。您可以通过其中一个 Amazon AWS 软件开发套件 (SDKs) 或 CloudWatch API 工具来使用 API。CloudWatch

下表列出了 CloudWatch 指标的一些常见用途。这些是入门建议，并不全面。

如何？	监控此指标
跟踪成功请求数	`SuccessfulRequestCount` 指标的 sum 统计数据
了解应用程序是否已达到其最大吞吐量	`ThrottledCount` 指标的 sum 统计数据
查找应用程序的响应时间	`ResponseTime` 指标的 average 统计数据
查找应用程序的错误数	`ServerErrorCount` 和 `UserErrorCount` 指标的 sum 统计数据
查找计费字符数	`CharacterCount` 指标的 sum 统计数据

您必须拥有相应的 CloudWatch 权限才能监控 Amazon Translate。 CloudWatch 有关更多信息，请参阅亚马逊 CloudWatch 用户指南 CloudWatch中的亚马逊身份验证和访问控制。

查看 Amazon Translate 指标

在 CloudWatch 控制台中查看 Amazon Translate 指标。

查看指标（CloudWatch 控制台）

1. 登录 AWS Management Console 并打开 CloudWatch 控制台，网址为https://console.aws.amazon.com/cloudwatch/。
2. 依次选择指标、全部指标和 AWS/Translate。
3. 选择维度、指标名称，然后选择添加到图表。
4. 选择日期范围的值。指定日期范围的指标计数将显示在该图表中。

使用记录 Amazon Translate API 调用 AWS CloudTrail

Amazon Translate 与 AWS CloudTrail一项服务集成，可记录 IAM 实体或 AWS 服务在 Amazon Translate 中采取的行动。 CloudTrail 将 Amazon Translate 的所有 API 调用捕获为事件。这包含来自 Amazon Translate 控制台的调用和对 Amazon Translate API 操作的代码调用。如果您创建 CloudTrail 跟踪，则可以将 CloudTrail 事件（包括 Amazon Translate 的事件）持续传输到亚马逊简单存储服务 (Amazon S3) 存储桶。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。您可以使用收集的信息 CloudTrail 来确定向 Amazon Translate 发出的请求、发出请求的 IP 地址、谁提出了请求、何时提出请求以及其他详细信息。

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

主题

- Amazon Translate 中的信息 CloudTrail
- 了解 Amazon Translate 日志文件条目

Amazon Translate 中的信息 CloudTrail

CloudTrail 在您创建 AWS 账户时已在您的账户上启用。当 Amazon Translate 中发生活动时，该活动会与其他 AWS 服务 CloudTrail 事件一起记录在事件历史记录中。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 Amazon Translate 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，在使用控制台创建跟踪时，此跟踪应用于所有 AWS 区域。跟踪记录 AWS 分区所有区域的事件，将日志文件传送至指定的 S3 存储桶。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：

- 创建跟踪记录概述
- 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 Amazon Translate 操作均由 API 参考部分记录 CloudTrail 并记录在案。例如，调用`ImportTerminology`和`TranslateText`操作会在 CloudTrail 日志文件中生成条目。`DeleteTerminology`

每个事件或日志条目都包含有关生成请求的人员信息。此信息可帮助您确定以下内容：

- 请求是使用根用户凭证还是根用户凭证发出的
- 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 Translate 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。事件代表来自任何来源的单个请求，包括有关请求的操作、操作的日期和时间、请求参数等的信息。 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`TranslateText`操作的 CloudTrail 日志条目。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:user/Administrator",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "Administrator"
    },
    "eventTime": "2019-09-03T20:32:50Z",
    "eventSource": "translate.amazonaws.com",
    "eventName": "TranslateText",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/1.16.207 Python/3.4.7
 Linux/4.9.184-0.1.ac.235.83.329.metal1.x86_64 botocore/1.12.197",
    "requestParameters": {
        "text": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "sourceLanguageCode": "en",
        "targetLanguageCode": "fr"
    },
    "responseElements": {
        "translatedText": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "sourceLanguageCode": "en",
        "targetLanguageCode": "fr"
    },
    "requestID": "f56da956-284e-4983-b6fc-59befa20e2bf",
    "eventID": "1dc75278-84d7-4bb2-861a-493d08d67391",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

CloudWatch Amazon Translate 的指标和维度

要监控解决方案的性能，请使用 Amazon Translate 的亚马逊 CloudWatch 指标和维度。

CloudWatch Amazon Translate 的指标

指标	描述
`CharacterCount`	请求中的计费字符数。 有效维度：语言对、操作 有效统计数据：平均值、最大值、最小值、总计 单位：计数
`ResponseTime`	响应请求所花费的时间。 有效维度：语言对、操作 有效统计数据：数据采样、平均值 单位：对于数据采样，计数。对于平均值统计数据，毫秒。
`ServerErrorCount`	服务器错误的数量。服务器错误的 HTTP 响应代码范围为 500 到 599。 有效维度：操作 有效统计数据：Average、Sum 单位：计数
`SuccessfulRequestCount`	成功翻译请求数。成功请求的响应代码为 200 至 299。 有效维度：操作 有效统计数据：Average、Sum 单位：计数

指标	描述
ThrottledCount	遭受节流的请求数。使用 ThrottledCount 确定您的应用程序向 Amazon Translate 发送请求的速度是否比您的账户被配置为接受请求的速度更快。有关更多信息，请参阅 https://docs.aws.amazon.com/general/latest/gr/aws_service_limits.html#limits_amazon_translate 中的Amazon Web Services 一般参考 Amazon Translate 限制。 有效维度：操作 有效统计数据：Average、Sum 单位：计数
UserErrorCount	发生的用户错误数。用户错误的 HTTP 响应代码范围为 400 到 499。 有效维度：操作 有效统计数据：Average、Sum 单位：计数

CloudWatch Amazon Translate 的尺寸

使用以下维度可筛选 Amazon Translate 指标。指标按源语言和目标语言分组。

维度	描述
LanguagePair	仅限包含指定语言的指标。
Operation	仅限包含指定操作的指标。

使用亚马逊监控 Amazon Translate 事件 EventBridge

Amazon Trans EventBridge late 与亚马逊集成，可通知您有关影响您的翻译任务和并行数据资源的更改。来自 AWS 服务的事件几乎实时 EventBridge 地传送到。您可以编写简单规则来指示您关注的事件，并指示要在事件匹配规则时执行的自动化操作。例如，可自动启动的操作包括：

- 调用函数 AWS Lambda

- 调用 AWS Systems Manager 运行命令
- 将事件中继到 Amazon Kinesis Data Streams
- 激活 AWS Step Functions 状态机
- 通知 Amazon SNS 主题或 Amazon SQS 队列

有关更多信息，请参阅 EventBridge 《亚马逊 EventBridge 用户指南》中的创建对事件做出反应的亚马逊规则。

Amazon Translate 事件

以下是来自 Amazon Translate 的示例事件。

批量翻译任务的事件

您可以使用 Amazon Translate 控制台或StartTextTranslationJob操作来运行批量翻译作业。在这些作业完成时，无论成功还是失败，Amazon Translate 都会发送事件。这些事件与以下示例类似。

```
{
 "version": "0",
 "id": "CWE-event-id",
 "detail-type": "Translate TextTranslationJob State Change",
 "source": "aws.translate",
 "account": "111122223333",
 "time": "2017-04-22T03:31:47Z",
 "region": "us-east-1",
 "resources": [],
 "detail": {
    "jobId": "01234567-0123-0123-0123-012345678901",
    "jobStatus": "STATUS"
 }
}
```

`jobStatus` 属性的值取决于接收 Amazon Translate 所发送事件的任务状态。`jobStatus` 值为：

- `COMPLETED` – 作业已成功完成，并且输出可用。
- `COMPLETED_WITH_ERROR` – 作业已完成，但出现错误。您可以在作业的输出中分析错误。
- `STOPPED` – 任务已停止。
- `FAILED` – 任务未完成。要获取详细信息，请使用DescribeTextTranslationJob操作。

并行数据资源的事件

在您使用 Amazon Translate 创建或更新并行数据资源时，该服务会发送一个事件来指示操作是成功还是失败。

您可以使用 Amazon Translate 控制台或CreateParallelData操作创建并行数据资源。在您执行此操作时，Amazon Translate 会发送如下所示的事件。

```
{
 "version": "0",
 "id": "CWE-event-id",
 "detail-type": "Translate Parallel Data State Change",
 "source": "aws.translate",
 "account": "111122223333",
 "time": "2017-04-22T03:31:47Z",
 "region": "us-east-1",
 "resources": [arn:aws:translate:us-east-1:111122223333:parallel-data/
ExampleParallelData],
 "detail": {
    "operation": "CreateParallelData",
    "name": "ExampleParallelData",
    "status": "STATUS"
 }
}
```

`status` 属性的值为：

- `ACTIVE` – `CreateParallelData` 操作成功，资源已准备就绪，可供您使用。
- `FAILED` – `CreateParallelData` 操作失败。

您可以使用 Amazon Translate 控制台或UpdateParallelData操作来更新并行数据资源。在您执行此操作时，Amazon Translate 会发送如下所示的事件。

```
{
 "version": "0",
 "id": "CWE-event-id",
 "detail-type": "Translate Parallel Data State Change",
 "source": "aws.translate",
 "account": "111122223333",
 "time": "2017-04-22T03:31:47Z",
 "region": "us-east-1",
```

```
 "resources": [arn:aws:translate:us-east-1:111122223333:parallel-data/
ExampleParallelData],
 "detail": {
    "operation": "UpdateParallelData",
    "name": "ExampleParallelData",
    "status": "STATUS",
    "latestUpdateAttemptStatus": "STATUS",
    "latestUpdateAttemptAt": "2017-04-22T03:31:47Z"
 }
}
```

`status` 属性提供并行数据资源先前版本的状态，该版本正在通过更新替换。值为：

- `ACTIVE` – 先前版本已成功创建或更新。
- `FAILED` – 创建或更新先前版本失败。

`latestUpdateAttemptStatus` 属性提供并行数据资源新版本的状态，该版本正在通过更新创建。值为：

- `ACTIVE` – `UpdateParallelData` 操作成功，更新后的资源已准备就绪，可供您使用。
- `FAILED` – `UpdateParallelData` 操作失败。

Amazon Translate 的合规性验证

作为多项合规计划的一部分，第三方审计师会评估 Amazon Translate 的安全与 AWS 合规性。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 Amazon Artifact 中下载报告。

您使用 Amazon Translate 的合规性责任取决于您数据的敏感度、贵公司的合规性目标以及适用的法律法规。 AWS 提供以下资源来帮助满足合规性：

- 《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 Translate 中的故障恢复能力

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 Translate 中的基础设施安全性

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

要通过网络访问 Amazon Translate，您可以使用 AWS 已发布的 API 调用。客户端必须支持 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。

Amazon Translate 和接口 VPC 端点 (AWS PrivateLink)

您可以创建接口 VPC 端点，在您的 VPC 和 Amazon Translate 之间建立私有连接。接口终端节点由一项技术提供支持 AWS PrivateLink，该技术使您 APIs 无需互联网网关、NAT 设备、VPN 连接或 Di AWS rect Connect 连接即可私密访问 Amazon Translate。您的 VPC 中的实例不需要公有 IP 地址即可与 Amazon Translate 通信 APIs。VPC 和 Amazon Translate 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《Amazon VPC 用户指南》中的接口 VPC 端点 (AWS PrivateLink)。

Amazon Translate VPC 端点注意事项

在为 Amazon Translate 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 Translate 支持从 VPC 调用它的所有 API 操作。

为 Amazon Translate 创建接口 VPC 端点

您可以使用亚马逊 VPC 控制台或 AWS Command Line Interface (AWS CLI) 为 Amazon Translate 服务创建 VPC 终端节点。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 Translate 创建 VPC 端点：

- com.amazonaws。*region*。翻译

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 Translate 发送 API 请求，例如 `translate.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 Translate 创建 VPC 端点策略

您可以为 VPC 端点附加控制对 Amazon Translate 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 Translate 实时翻译操作的 VPC 端点策略

下面是用于 Amazon Translate 中实时翻译的端点策略示例。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列出的 Amazon Translate 操作的访问权限。

```
{
```

```
            "Statement":[
                {
                    "Principal":"*",
                    "Effect":"Allow",
                    "Action":[
                        "translate:TranslateText",
                    ],
                    "Resource":"*"
                }
            ]
        }
```

示例：Amazon Translate 批量翻译操作的 VPC 端点策略

下面是用于 Amazon Translate 中批量翻译的端点策略示例。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列出的 Amazon Translate 操作的访问权限。

```
{
            "Statement":[
                {
                    "Principal":"*",
                    "Effect":"Allow",
                    "Action":[
                        "translate:StartTextTranslationJob",
                        "iam:PassRole"
                    ],
                    "Resource":"*"
                }
            ]
        }
```

准则和限额

以下各节包含有关 Amazon Translate 准则和限额的信息。

主题

- 支持的 AWS 区域
- 合规
- 节流
- 指南
- 服务限额

支持的 AWS 区域

有关支持 Amazon Translate 的 AWS 地区列表，请参阅《AWS 一般参考》中的 Amazon Translate 终端节点和配额。

合规

有关 Amazon Translate 合规性计划的更多信息，请参阅AWS 合规性、AWS 合规性计划和合规性计划范围内的AWS 服务。

节流

Amazon Translate 可以扩展以便处理客户运营流量。如果您持续遇到节流问题，请联系 AWS Support。

指南

为持续改进分析模型的质量，Amazon Translate 可能会存储您的数据。要了解更多信息，请参阅 Amazon Translate 常见问题。

您可以通过联系 AWS Support 来请求我们删除您的数据，并请求不存储与您的账户关联的未来数据。但是，由于删除您的数据也可能删除有助于提高翻译质量的唯一训练数据，这样做可能会降低您的翻译质量。

服务限额

Amazon Translate 有以下服务准则和限额。

同步实时翻译限额

描述	限制
字符编码	UTF-8
最大输入文本大小	10000 字节
每个文档的最大字符数	100000
最大文档大小	100000 字节

异步批量翻译限额

描述	限制
字符编码	UTF-8
每个文档的最大字符数	1000000
每个文档的最大大小	20 MB
单个文档中可翻译文本的最大大小	1 MB
批处理作业请求中目标语言的最大数量	10
批处理中文档的最大数量	1000000
批处理中总文档的最大大小	5 GB
并发批量翻译任务的最大数量	10
排队批量翻译任务的最大数量	1000
StartTextTranslationJob API 操作的每秒事务数	5
DescribeTextTranslationJob API 操作的每秒事务数	10

描述	限制
ListTextTranslationJobs API 操作的每秒事务数	10
StopTextTranslationJob API 操作的每秒事务数	5

自定义术语限额

描述	限制
最大自定义术语文件大小	10MB
每 AWS 区域每个 AWS 账户的最大自定义术语文件数	100
每个自定义术语文件的最大目标语言数	10
每个自定义术语的最大源和目标文本长度	200 字节
每个 `TranslateText` 或 `StartTextTranslationJob` 请求的最大术语文件数。	1
ImportTerminology API 操作的每秒事务数	5
GetTerminology API 操作的每秒事务数	10
ListTerminologies API 操作的每秒事务数	10
DeleteTerminology API 操作的每秒事务数	5

并行数据限额

描述	限制
每 AWS 区域每个 AWS 账户的最大并行数据资源数	1000
并行数据输入文件的最大大小	5 GB
并行数据资源中源语言的最大数量	1
并行数据输入文件中单个句段或记录的最大大小	1000 字节

描述	限制
并行数据资源的最大并发创建或更新操作数	1
CreateParallelData API 操作的每秒事务数	5
GetParallelData API 操作的每秒事务数	10
ListParallelData API 操作的每秒事务数	10
UpdateParallelData API 操作的每秒事务数	5
DeleteParallelData API 操作的每秒事务数	5

Amazon Translate 文档历史记录

下表介绍了此版本的 Amazon Translate 的文档。

变更	说明	日期
新功能：简洁	Translate 现在支持实时文本翻译的简洁性。简洁可缩短大多数翻译的翻译输出的长度（与不简洁的翻译输出相比）。有关更多信息，请参阅在 Amazon Translate 中使用简洁性。	2023 年 10 月 31 日
用于实时翻译输入文档的语言自动检测	现在，您可以在输入文档进行实时翻译（控制台或 API）时使用语言自动检测。有关更多信息，请参阅实时翻译。	2023 年 8 月 3 日
Word (.docx) 文件作为输入进行实时翻译	现在，您可以使用 .docx 文件（除文本文件和 HTML 文件之外）作为输入进行实时翻译（控制台或 API）。有关更多信息，请参阅实时翻译。	2023 年 7 月 17 日
自定义术语的增强	Translate 现在支持增强的自定义术语特征，可提高翻译的流畅度和准确性。有关更多信息，请参阅使用自定义术语自定义翻译。	2023 年 6 月 30 日
文本或 HTML 文件作为输入进行实时翻译	现在，您可以使用文本文件或 HTML 文件作为输入进行实时翻译（控制台或 API）。有关更多信息，请参阅实时翻译。	2023 年 5 月 23 日
TranslateReadOnly 策略中允许的新操作	Amazon Translate 现在允许在 TranslateReadOnly 托管	2023 年 5 月 23 日

	策略中执行该`Translate Document` 操作。有关更多信息，请参阅 AWS 托管策略：TranslateReadOnly。	
Translate 现在支持在其他区域进行异步批处理。	Translate 现在支持在其他区域进行异步批处理。有关更多信息，请参阅使用 Amazon Translate 进行异步批处理。	2023 年 3 月 28 日
实时翻译的输入大小增加	现在，您最多可以输入多达 10000 个字符进行实时翻译。有关更多信息，请参阅 Amazon Translate（控制台）入门。	2022 年 12 月 16 日
支持批处理模式下的嵌套输入文件夹	现在，您可以使用嵌套输入文件夹进行批量翻译作业。有关更多信息，请在 Amazon Translate 中参阅运行批量翻译作业。	2022 年 11 月 18 日
支持批处理模式下的自动语言检测	现在，您可以在批量翻译作业中自动检测源语言。因此，您现在可以在批量翻译作业中输入不同源语言的文档。有关更多信息，请在 Amazon Translate 中参阅运行批量翻译作业。	2022 年 11 月 18 日
支持多种目标语言	现在，您可以在批量翻译作业中指定多种目标语言。有关更多信息，请在 Amazon Translate 中参阅运行批量翻译作业。	2022 年 10 月 10 日

支持标签	现在，您可以在 Amazon Translate 中标记ParallelData和自定义术语资源。有关更多信息，请参阅在 Amazon Translate 标记您的资源。	2022 年 10 月 6 日
持其他语言的正式性设置	现在，您可以在 Amazon Translate 中为荷兰语、韩语和墨西哥西班牙语设置翻译的正式程度。有关更多信息，请参阅在 Amazon Translate 设置正式性。	2022 年 10 月 5 日
单独的 API 参考	Amazon Translate API 参考现在是一个单独的文档。有关更多信息，请参阅 Amazon Translate API 参考。	2022 年 8 月 25 日
新特征	现在，您可以为翻译输出设置正式程度。有关更多信息，请参阅在 Amazon Translate 设置正式性。	2022 年 2 月 22 日
新特征	现在，您可以掩蔽翻译输出中的污言秽语。有关更多信息，请参阅在 Amazon Translate 中掩蔽污言秽语。	2021 年 11 月 24 日
AWS PrivateLink 支持	现在，您可以使用在您的 VPC 和 Amazon Translate 之间建立私有连接 AWS PrivateLink。有关更多信息，请参阅 Amazon Translate 和接口 VPC 端点 (AWS PrivateLink)。	2021 年 11 月 24 日

并行数据更新	现在，您可以使用 Amazon Translate 支持的任何语言创建并行数据资源。不再需要使用英语作为其中一种语言。有关并行数据的更多信息，请参阅使用并行数据自定义您的翻译（活动自定义翻译）。	2021 年 11 月 15 日
自定义术语方向性	现在，您可以创建多向术语，其中任何语言都可以作为源语言或目标语言。有关更多信息，请参阅创建自定义术语。	2021 年 11 月 11 日
新语言	Amazon Translate 现在支持以下语言：爱尔兰语、马拉地语、葡萄牙语（葡萄牙）和旁遮普语。如需查看 Amazon Translate 支持的所有语言，请参阅支持的语言和语言代码。	2021 年 11 月 10 日
新自定义加密设置	现在，您可以使用自己在 AWS Key Management Service中管理的客户托管式密钥对您的翻译输出进行加密。有关更多信息，请参阅运行批量翻译作业	2021 年 11 月 5 日
新文件格式支持	Amazon Translate 现在支持对 XML 本地化交换文件格式 (XLIFF) 文件进行异步批处理。有关所有支持的格式，请参阅支持的文件格式。	2021 年 6 月 9 日

EventBridge 整合	Amazon Translate 现在会 EventBridge 向亚马逊发送事件，通知您有关影响您的翻译任务和并行数据资源的更改。有关更多信息，请参阅使用亚马逊监控 Amazon Translate 事件 EventBridge。	2021 年 6 月 4 日
新限额	Amazon Translate 现在支持多达 1000 个排队批量翻译作业。有关所有 Amazon Translate 限额，请参阅指南和限制。	2021 年 4 月 23 日
增加限额	并行数据输入文件的最大大小已从 1 MB 增加到 5 MB。有关所有 Amazon Translate 限额，请参阅指南和限制。	2021 年 3 月 31 日
新语言	Amazon Translate 现在支持以下语言：亚美尼亚语、加泰罗尼亚语、波斯语、菲律宾塔加洛语、古吉拉特语、海地克里奥尔语、冰岛语、卡纳达语、哈萨克语、立陶宛语、马其顿语、马来亚拉姆语、马耳他语、蒙古语、僧伽罗语、泰卢固语、乌兹别克语和威尔士语。如需查看 Amazon Translate 支持的所有语言，请参阅支持的语言和语言代码。	2020 年 11 月 23 日

新特征	现在，您可以使用并行数据（由源文本及其翻译的示例组成）来自定义批量翻译作业。使用并行数据的作业称为活动自定义翻译作业。在这些作业中，Amazon Translate 会调整翻译输出以反映并行数据中的示例。有关更多信息，请参阅使用并行数据自定义您的翻译（活动自定义翻译）。	2020 年 11 月 23 日
新文件格式支持	Amazon Translate 现在支持以下 Office Open XML 文件格式作为异步批处理的输入：Word 文档 (.docx)、 PowerPoint 演示文稿 (.pptx)、Excel 工作簿 (.xlsx)。有关更多信息，请参阅开始批量翻译作业。	2020 年 7 月 29 日
新语言	Amazon Translate 现在支持西班牙语（墨西哥）的翻译。如需查看支持的所有语言，请参阅支持的语言和语言代码。	2020 年 4 月 30 日
新区域	Amazon Translate 支持在欧洲地区（伦敦）区域进行异步批处理。如需查看提供异步批处理的所有 AWS 区域，请参阅区域可用性。	2020 年 4 月 20 日
新特征	Amazon Translate 添加异步批量翻译功能。有关详细信息，请参阅异步批处理。	2019 年 12 月 23 日

新区域	Amazon Translate 增加了对亚太地区（香港）、亚太地区（悉尼）、欧洲（伦敦）、欧洲（巴黎）、欧洲（斯德哥尔摩）和美国西部（北加利福尼亚）区域的支持。有关 Amazon Translate 支持的 AWS 区域的完整列表，请参阅《Amazon Web Services 一般参考》中的 AWS 区域表或 AWS 区域和端点。	2019 年 11 月 25 日
新语言	Amazon Translate 新增以下语言的翻译：南非荷兰语、阿尔巴尼亚语、阿姆哈拉语、阿塞拜疆语、孟加拉语、波斯尼亚语、保加利亚语、加拿大法语、克罗地亚语、达里语、爱沙尼亚语、格鲁吉亚语、豪萨语、拉脱维亚语、普什图语、塞尔维亚语、斯洛伐克语、斯洛文尼亚语、索马里语、斯瓦希里语、塔加洛语和泰米尔语。有关 Amazon Translate 可直接翻译的语言组合的列表，请参阅支持的语言。	2019 年 11 月 25 日
新语言	Amazon Translate 新增以下语言的翻译：希腊语、匈牙利语、罗马尼亚语、泰语、乌克兰语、乌尔都语和越南语。有关 Amazon Translate 可直接翻译的语言组合的列表，请参阅支持的语言。	2019 年 10 月 3 日

新特征	Amazon Translate 新增 FedRAMP 合规性。有关更多信息，请参阅 合规。	2019 年 7 月 31 日
新特征	Amazon Translate 新增 SOC 合规性。有关更多信息，请参阅 合规。	2019 年 5 月 30 日
新区域	Amazon Translate 增加了对亚太地区（孟买）、亚太地区（新加坡）、亚太地区（东京）、亚太地区（悉尼）和加拿大（中部）区域的支持。有关 Amazon Translate 支持的 AWS 区域的完整列表，请参阅《Amazon Web Services 一般参考》中的 AWS 区域表或 AWS 区域和端点。	2019 年 5 月 8 日
新语言	Amazon Translate 新增以下语言的翻译：印地语、马来语、挪威语和波斯语。有关 Amazon Translate 可直接翻译的语言组合的列表，请参阅支持的语言。	2019 年 5 月 6 日
新区域	Amazon Translate 增加了对欧洲（法兰克福）和亚太地区（首尔）区域的支持。有关 Amazon Translate 支持的 AWS 区域的完整列表，请参阅《Amazon Web Services 一般参考》中的 AWS 区域表或 AWS 区域和端点。	2019 年 2 月 28 日

新特征	Amazon Translate 新增 PCI 合规性。有关更多信息，请参阅合规。	2018 年 12 月 12 日
新特征	Amazon Trans APIs late 增加了四项新的自定义术语功能，让您可以更好地控制自己的翻译。通过使用自定义术语和翻译请求，您可以确保您的品牌名称、角色名称、型号名称和其他独特内容每次都完全按照您的需要进行翻译，无论标准翻译或上下文如何。有关更多信息，请参阅自定义术语。	2018 年 11 月 27 日
新语言	Amazon Translate 现在可以翻译以下语言的文档：丹麦语、荷兰语、芬兰语、希伯来语、印度尼西亚语、韩语、波兰语和瑞典语。Amazon Translate 通过显著减少不支持的语言对的数量，持续改进其直接翻译。有关 Amazon Translate 可直接翻译的语言组合，请参阅支持的语言。	2018 年 11 月 20 日
新特征	Amazon Translate 增加了英语以外的受支持语言之间的直接翻译。有关 Amazon Translate 可直接翻译的语言组合，请参阅支持的语言。	2018 年 10 月 29 日
新特征	Amazon Translate 新增 HIPAA 合规性。有关更多信息，请参阅 合规。	2018 年 10 月 25 日

新特征	Amazon Translate 新增多种新语言的翻译：繁体中文、捷克语、意大利语、日语、俄语和土耳其语。有关 Amazon Translate 支持的语言的列表，请参阅支持的语言。	2018 年 7 月 17 日
新特征	Amazon Translate 增加了对自动源语言检测的支持。有关更多信息，请参阅 Amazon Translate 的工作原理。	2018 年 4 月 4 日
新指南	本指南是 Amazon Translate 开发人员指南的第一个版本。	2017 年 11 月 29 日

API 参考

Amazon Translate API 参考现在是一个单独的文档。有关更多信息，请参阅 Amazon Translate API 参考。

AWS 词汇表

有关最新 AWS 术语，请参阅《AWS 词汇表 参考资料》中的AWS 词汇表。

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。