



Guia do Desenvolvedor

Amazon Transcribe



Amazon Transcribe: Guia do Desenvolvedor

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens comerciais da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

Table of Contents

O que é o Amazon Transcribe?	1
Amazon Transcribe elegibilidade à HIPAA	1
Preços	2
Disponibilidade e cotas da região	2
Recursos disponíveis	4
Idiomas oferecidos	7
Linguagens de programação compatíveis	17
Conjuntos de caracteres	18
Abkhaz	21
Africâner	24
Árabe	25
Asturiano	26
Azerbaijano	26
Armênio	27
Bashkir	28
Basco	31
Bielorrusso	31
Bengali	32
Bósnio	34
Búlgaro	35
Catalão	36
Curdo central	37
Chinês simplificado	38
Chinês tradicional	39
Croata	40
Tcheco	41
Dinamarquês	41
Holandês	42
Inglês	43
Estoniano	43
Persa	44
Finlandês	45
Francês	46
Galego	47

Georgiano	47
Alemão	49
Grego	49
Gujarati	51
Hauçá	53
Hebraico	53
Hindi	54
Húngaro	56
Islandês	57
Indonésio	58
Italiano	58
Japonês	59
Cabila	59
Kannada	60
Cazaque	62
Quiníaruanda	63
Coreano	64
Quirguiz	64
Letão	67
Lituano	67
Luganda	68
Macedônio	68
Malaio	71
Malaiala	71
Maltês	73
Marathi	74
Mari das campinas	76
Mongol	78
Norueguês Bokmål	81
Oriá	81
Pachto	83
Polonês	85
Português	86
Punjabi	87
Romeno	89
Russo	90

Sérvio	91
Cingalês	93
Eslovaco	95
Esloveno	96
Somali	97
Espanhol	97
Sudanês	98
Suaíli	99
Sueco	99
Tagalo/filipino	100
Tâmil	100
Tatárico	102
Telugo	104
Tailandês	106
Turco	108
Ucraniano	109
Uigur	110
Uzbeque	113
Vietnamita	114
Galês	118
Uolofe	119
Zulu	120
Como funciona	121
Entrada e saída de dados	121
Formatos de mídia	122
Canais de áudio	123
Taxas de amostragem	124
Resultado	124
Transcrevendo números	127
Conceitos básicos	133
Cadastrar-se em uma Conta da AWS	134
Instalando os SDKs AWS CLI e	134
Configurar IAM credenciais	135
Criação de um Amazon S3 bucket	135
Criar uma política do IAM	136
Transcrevendo com o AWS Management Console	138

Transcrevendo com oAWS CLI	147
Iniciando um novo trabalho de transcrição	147
Obter o status de um trabalho de transcrição	149
Listando seus trabalhos de transcrição	150
Excluindo seu trabalho de transcrição	150
Transcrevendo com os SDKs AWS	151
Trabalhando com AWS SDKs	163
Transcrevendo com HTTP ou WebSockets	164
Transcrições de streaming	167
Práticas recomendadas	168
Streaming e resultados parciais	169
Estabilização de resultados parciais	170
Configurar uma transcrição de streaming	174
Codificação de fluxo de eventos	189
Quadros de dados	192
Fila de Job	193
Habilitação de trabalhos em lote	193
Marcar recursos	198
Controle de acesso com base em tags	199
Adicionar etiquetas aosAmazon Transcribe recursos do	200
Separação de locutores (diarização)	204
Separar locutores em uma transcrição em lote	205
Separar locutores em uma transcrição de streaming	208
Exemplo de saída	211
Transcrevendo áudio multicanal	217
Usando a identificação de canais em uma transcrição em lote	218
Usando a identificação de canais em uma transcrição de streaming	222
Exemplo de saída	223
Identificar idiomas	231
Identificação de idioma em lote	231
Identificar idiomas em áudio com vários idiomas	232
Melhorar a precisão da identificação de idiomas	233
Utilizar a identificação de idioma com outros recursos do Amazon Transcribe	234
Usar a identificação de idioma com transcrições em lote	235
Identificação do idioma de streaming	242
Identificação de idiomas em áudio com vários idiomas	243

Usar a identificação de idioma com streaming de mídia	243
Transcrições alternativas	251
Solicitando transcrições alternativas	253
Melhorar a precisão da transcrição	258
Vocabulários personalizados	259
Tabelas de vocabulário personalizadas versus listas	260
Criar um vocabulário personalizado usando uma tabela	261
Criação de um vocabulário personalizado usando uma lista	272
Usar um vocabulário personalizado	275
Modelos de linguagem personalizados	282
Fontes de dados	282
Treinamento versus ajuste de dados	283
Criação de um modelo de linguagem personalizado	284
Usando um modelo de linguagem personalizado	290
Filtrando palavras	298
Criação de um filtro de vocabulário	299
Criação de filtros de vocabulário personalizados	300
Usando um filtro de vocabulário personalizado	305
Usando um filtro de vocabulário personalizado em uma transcrição em lote	275
Usando um filtro de vocabulário personalizado em uma transcrição de streaming	279
Detectando fala tóxica	314
Usando a detecção de fala tóxica	315
Usando detecção de fala tóxica em uma transcrição em lote	315
Exemplo de saída	320
Redação de transcrições	322
Editar PII no trabalho em lote	323
Editar ou identificar PII em um fluxo em tempo real	330
Exemplo de saída	336
Exemplo de saída editada (lote)	336
Exemplo de saída de streaming editada	339
Exemplo de saída de identificação PII	340
Criar legendas	343
Geração de arquivos de legendas	344
Analisar o áudio da central de atendimento	347
Casos de uso comuns	347
Considerações e informações adicionais	349

Disponibilidade e cotas da região	350
Análise pós-chamada	351
Insights pós-chamada	351
Criar categorias	354
Iniciar uma transcrição	367
Saída da análise pós-chamada	376
Habilitar o resumo generativo de chamada	389
Análise de chamadas em tempo real	394
Insights em tempo real	395
Criar categorias	397
Análise pós-chamada com transcrições em tempo real	404
Iniciar uma transcrição	412
Saída de análise de chamadas em tempo real	421
Transcrevendo Amazon Chime chamadas	427
Exemplos de código	429
Ações	430
CreateVocabulary	431
DeleteMedicalTranscriptionJob	434
DeleteTranscriptionJob	437
DeleteVocabulary	441
GetTranscriptionJob	443
GetVocabulary	446
ListMedicalTranscriptionJobs	449
ListTranscriptionJobs	454
ListVocabularies	460
StartMedicalTranscriptionJob	464
StartStreamTranscriptionAsync	476
StartTranscriptionJob	480
UpdateVocabulary	499
Cenários	502
Criar e refinar um vocabulário personalizado	503
Transcrever áudio e obter dados do trabalho	513
Exemplos entre serviços	524
Criar uma aplicação Amazon Transcribe	524
Criar uma aplicação de transmissão do Amazon Transcribe	525
Converter texto em fala e de volta em texto	525

Segurança	527
Identity and Access Management	528
Público	528
Autenticando com identidades	529
Gerenciando acesso usando políticas	532
Como Amazon Transcribe funciona com o IAM	535
Prevenção de ‘confused deputy’	543
Exemplos de políticas baseadas em identidade	544
Solução de problemas	553
Proteção de dados	555
Privacidade do tráfego entre redes	556
Criptografia de dados	556
Optar por não usar seus dados para melhorar o serviço	559
Monitoramento Amazon Transcribe	559
Monitoramento com CloudWatch	560
Monitoramento Amazon Transcribe com CloudTrail	561
Usando Amazon EventBridge com Amazon Transcribe	565
Validação de conformidade	572
Resiliência	574
Segurança da infraestrutura	574
Análise e gerenciamento de vulnerabilidades	575
VPC endpoints (AWS PrivateLink)	575
Sub-redes compartilhadas	577
Práticas recomendadas de segurança	577
Amazon Transcribe Medical	580
Disponibilidade e cotas da região	581
Especialidades médicas	583
Transcrevendo termos e medidas médicas	584
Transcrevendo números	586
Transcrevendo uma conversa médica	588
Transcrevendo um arquivo de áudio	589
Transcrever um stream em tempo real	594
Habilitar particionamento de alto-falantes	597
Transcrevendo áudio multicanal	607
Transcrevendo um ditado médico	615
Transcrevendo um arquivo de áudio	616

Transcrevendo um ditado médico em streaming	621
Criação e uso de vocabulários médicos personalizados	623
Criando um arquivo de texto para seu vocabulário médico personalizado	624
Usando um arquivo de texto para criar um vocabulário médico personalizado	629
Transcrevendo um arquivo de áudio usando um vocabulário médico personalizado	631
Transcrevendo um stream em tempo real usando um vocabulário médico personalizado	633
Conjunto de caracteres para Amazon Transcribe Medicina	636
Identificando PHI em uma transcrição	637
Identificando PHI em um arquivo de áudio	639
Identificando PHI em um stream em tempo real	643
Gerando transcrições alternativas	645
VPC endpoints (AWS PrivateLink)	648
Considerações para Amazon Transcribe Endpoints médicos de VPC	648
Criação de um endpoint VPC de interface para Amazon Transcribe Médico	648
Criação de uma política de endpoint de VPC para Amazon Transcribe Streaming médico	649
Sub-redes compartilhadas	650
AWS HealthScribe	651
Arquivo de transcrição	653
Arquivo de documentação clínica	653
Iniciar um trabalho do AWS HealthScribe	655
Exemplo de saída	657
Criptografia de dados em repouso para o AWS HealthScribe	670
Criação de uma chave gerenciada pelo cliente	671
Especificar uma chave gerenciada pelo cliente para o AWS HealthScribe	672
Contexto de criptografia do AWS KMS	673
Histórico do documento	674
AWS Glossário	685
.....	dclxxxvi

O que é o Amazon Transcribe?

Amazon Transcribe é um serviço de reconhecimento automático de fala que usa modelos de aprendizado de máquina para converter áudio em texto. Você pode usar Amazon Transcribe como um serviço de transcrição independente ou para adicionar speech-to-text recursos a qualquer aplicativo.

Com o Amazon Transcribe, você pode melhorar a precisão de seu caso de uso específico com a personalização do idioma, filtrar o conteúdo para garantir a privacidade do cliente ou o idioma adequado ao público, analisar o conteúdo em áudio multicanal, particionar a fala de palestrantes individuais e muito mais.

Você pode transcrever mídia em tempo real (streaming) ou transcrever arquivos de mídia localizados em um Amazon S3 bucket (lote). Para ver quais idiomas são compatíveis com cada tipo de transcrição, consulte a [Idiomas oferecidos e recursos específicos do idioma](#) tabela.

Tópicos

- [Amazon Transcribe elegibilidade à HIPAA](#)
- [Preços](#)
- [Disponibilidade e cotas da região](#)

Veja [o que é Amazon Transcribe?](#) para um breve tour em vídeo deste serviço.

Para saber mais, consulte [Como o Amazon Transcribe funciona](#) e [Conceitos básicos do Amazon Transcribe](#).

Tip

As informações sobre a Amazon Transcribe API estão localizadas na [Referência da API](#).

Amazon Transcribe elegibilidade à HIPAA

Amazon Transcribe AWS está coberto pela elegibilidade HIPAA e pela BAA, que exige que os clientes da BAA criptografem todas as PHI em repouso e em trânsito quando em uso. A identificação automática de PHI está disponível sem custo adicional e em todas as regiões onde Amazon Transcribe opera. Para obter mais informações, consulte a [elegibilidade para a HIPAA](#) e o BAA.

Preços

Amazon Transcribe é um pay-as-you-go serviço; o preço é baseado em segundos de áudio transcrito, cobrados mensalmente.

O uso é cobrado em incrementos de um segundo, com uma cobrança mínima de 15 segundos. Observe que cobranças adicionais se aplicam a recursos como redação de conteúdo de PII e modelos de linguagem personalizados.


Para obter informações sobre o custo de cada uma das Regiões da AWS, consulte [Amazon Transcribe Preços](#).

Disponibilidade e cotas da região

Amazon Transcribe é suportado nas seguintes Regiões da AWS:

Região	Tipo de transcrição
af-south-1 (Cidade do Cabo)	em lote de lote de lote de
ap-east-1 (Hong Kong)	lote
ap-northeast-1 (Tóquio)	em lote de lote de lote de
ap-northeast-2 (Seul)	em lote de lote de lote de
ap-south-1 (Mumbai)	em lote de lote de lote de
ap-southeast-1 (Cingapura)	em lote de lote de lote de
ap-southeast-2 (Sydney)	em lote de lote de lote de
ca-central-1 (Canadá, Central)	em lote de lote de lote de
eu-central-1 (Frankfurt)	em lote de lote de lote de
eu-north-1 (Estocolmo)	lote
eu-west-1 (Irlanda)	em lote de lote de lote de
eu-west-2 (Londres)	em lote de lote de lote de

Região	Tipo de transcrição
eu-west-3 (Paris)	lote
me-south-1 (Bahrein)	lote
sa-east-1 (São Paulo)	em lote de lote de lote de
us-east-1 (Norte da Virgínia)	em lote de lote de lote de
us-east-2 (Ohio)	em lote de lote de lote de
us-gov-east-1 (GovCloud, Leste dos EUA)	em lote de lote de lote de
us-gov-west-1 (GovCloud, Oeste dos EUA)	em lote de lote de lote de
us-west-1 (São Francisco)	lote
us-west-2 (Oregon)	em lote de lote de lote de

 Important

O suporte regional difere para Amazon Transcribe [Amazon Transcribe Medical](#), e [Call Analytics](#).

Para obter os endpoints de cada região suportada, consulte [Endpoints de serviço](#) na Referência AWSgeral.

Para obter uma lista de cotas relacionadas às suas transcrições, consulte as [cotas de serviço na Referência geral](#). AWS Algumas cotas podem ser alteradas mediante solicitação. Se a coluna Ajustável contiver “Sim”, você poderá solicitar um aumento. Para fazer isso, selecione o link fornecido.

Amazon Transcribe features


Para ajudá-lo a decidir qual Amazon Transcribe solução é mais adequada ao seu caso de uso, a tabela a seguir oferece uma comparação de recursos.

Observe que “lote” e 'post-call' se referem à transcrição de um arquivo localizado em um bucket do Amazon S3 e “streaming” e 'real-time' se referem à transcrição de mídia em tempo real.

Atributo	Amazon Transcribe	Amazon Transcribe Medical ¹	Amazon Transcribe Análise de chamadas
Opções de configuração			
Transcrições alternativas	lote de streaming	lote de streaming	não
Identificação de canal	lote de streaming	lote de streaming	post-call, real-time
Enfileiramento de tarefas	batch (lote)	não	post-call
Identificação de idioma	lote de streaming	não	post-call
Identificação de vários idiomas	lote de streaming	não	não
Diarização de locutores	lote de streaming	lote de streaming	post-call
Transcrever dígitos ²	lote de streaming	lote de streaming	post-call, real-time
Análise de conversas			
Características da chamada	não	não	post-call
Resumo de chamadas ²	não	não	post-call

Atributo	Amazon Transcribe	Amazon Transcribe Medical ¹	Amazon Transcribe Análise de chamadas
Categorização personalizada	não	não	post-call
Eventos da categoria em tempo real	não	não	real-time
Detecção de problemas em tempo real ²	não	não	real-time
Sentimento do locutor em tempo real	não	não	real-time
Sentimento do locutor	não	não	post-call
Personalização do idioma			
Modelos de linguagem personalizados ²	lote de streaming	não	post-call, real-time
Vocabulários personalizados	lote de streaming	lote de streaming	post-call, real-time
Organização de recursos			
Tags	batch (lote)	batch (lote)	post-call
Dados sigilosos			
Identificar informações de saúde pessoais ²	não	lote de streaming	não
Identificação de informações de identificação pessoal ²	streaming	não	real-time

Atributo	Amazon Transcribe	Amazon Transcribe Medical ¹	Amazon Transcribe Análise de chamadas
Edição de áudio ²	não	não	post-call, real-time
Editar transcrições ²	lote de streaming	não	post-call, real-time
Filtragem de vocabulário	lote de streaming	não	post-call, real-time
Vídeo			
Legendas	batch (lote)	não	não

 ¹ Amazon Transcribe Medical só está disponível em inglês dos EUA.

² Esse recurso não está disponível para todos os idiomas. Consulte mais detalhes na tabela [Idiomas oferecidos e recursos específicos do idioma](#).

Idiomas oferecidos e recursos específicos do idioma

Os idiomas suportados pelo Amazon Transcribe estão listados na tabela a seguir; também estão listados os recursos específicos do idioma. Verifique se o recurso que você deseja usar é compatível com o idioma em sua mídia antes de continuar com a transcrição.

Para ver a lista completa de Amazon Transcribe recursos, consulte o [resumo dos recursos](#).

Na tabela a seguir, 'lote' se refere à transcrição de um arquivo de mídia localizado em um Amazon S3 bucket e 'streaming' se refere à transcrição de mídia transmitida em tempo real. Para transcrições do Call Analytics, 'post-call' refere-se à transcrição de um arquivo de mídia localizado em um Amazon S3 bucket e 'real-time' se refere à transcrição de mídia transmitida em tempo real.

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Abkhaz	ab-GE	batch (lote)	não	batch (lote)	não	não	não
Africâner	af-ZA	batch (lote)	não	batch (lote)	não	não	não
Árabe, Golfo	ar-AE	batch (lote)	não	não	não	não	post-call
Árabe, padrão moderno	ar-SA	batch (lote)	não	não	não	não	não
Armênio	hy-AM	batch (lote)	não	batch (lote)	não	não	não
Asturiano	ast-ES	batch (lote)	não	batch (lote)	não	não	não

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Azerbaijano	az-AZ	batch (lote)	não	batch (lote)	não	não	não
Bashkir	ba-RU	batch (lote)	não	batch (lote)	não	não	não
Basco	eu-ES	batch (lote)	não	batch (lote)	não	não	não
Bielorrusso	be-BY	batch (lote)	não	batch (lote)	não	não	não
Bengali	bn-IN	batch (lote)	não	batch (lote)	não	não	não
Bósnio	bs-BA	batch (lote)	não	batch (lote)	não	não	não
Búlgaro	bg-BG	batch (lote)	não	batch (lote)	não	não	não
Catalão	ca-ES	batch (lote)	não	batch (lote)	não	não	não
Curdo central, Irã	ckb-IR	batch (lote)	não	batch (lote)	não	não	não
Curdo central, Iraque	ckb-IQ	batch (lote)	não	batch (lote)	não	não	não

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Chinês simplificado	zh-CN	lote de streaming	não	não	não	não	post-call
Chinês tradicional	zh-TW	batch (lote)	não	não	não	não	não
Croata	hr-HR	batch (lote)	não	batch (lote)	não	não	não
Tcheco	cs-CZ	batch (lote)	não	batch (lote)	não	não	não
Dinamarquês	da-DK	batch (lote)	não	batch (lote)	não	não	não
Holandês	nl-NL	batch (lote)	não	batch (lote)	não	não	não
Inglês, australia no	en-AU	lote de streaming	lote de streaming	lote de streaming	lote de streaming	streaming	post-call, real-time
Inglês, britânico	en-GB	lote de streaming	lote de streaming	lote de streaming	lote de streaming	streaming	post-call, real-time
Inglês, indiano	en-IN	batch (lote)	batch (lote)	batch (lote)	não	não	post-call
Inglês, irlandês	en-IE	batch (lote)	batch (lote)	batch (lote)	não	não	post-call

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Inglês , Nova Zelândia	en-NZ	batch (lote)	batch (lote)	batch (lote)	não	não	não
Inglês , escocês	en-AB	batch (lote)	batch (lote)	batch (lote)	não	não	post-call
Inglês , África do Sul	en-ZA	batch (lote)	batch (lote)	batch (lote)	não	não	não
Inglês , EUA	en-US	lote de streaming	lote de streaming	lote de streaming	lote de streaming	lote de streaming	post-call, real-time
Inglês , galês	en-WL	batch (lote)	batch (lote)	batch (lote)	não	não	post-call
Estoniano	et-ET	batch (lote)	não	batch (lote)	não	não	não
Persa	fa-IR	batch (lote)	não	não	não	não	não
Finlandês	fi-FI	batch (lote)	não	batch (lote)	não	não	não
Francês	fr-FR	lote de streaming	não	lote de streaming	não	não	post-call, real-time
Francês , canadense	fr-CA	lote de streaming	não	lote de streaming	não	não	post-call, real-time

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Galego	gl-ES	batch (lote)	não	batch (lote)	não	não	não
Georgiano	ka-GE	batch (lote)	não	batch (lote)	não	não	não
Alemão	de-DE	lote de streaming	lote de streaming	lote de streaming	lote de streaming	não	post-call, real-time
Alemão, Suíça	de-CH	batch (lote)	batch (lote)	batch (lote)	não	não	post-call
Grego	el-GR	batch (lote)	não	batch (lote)	não	não	não
Gujarati	gu-IN	batch (lote)	não	batch (lote)	não	não	não
Hauçá	ha-NG	batch (lote)	não	batch (lote)	não	não	não
Hebraico	he-IL	batch (lote)	não	não	não	não	não
Hindi, indiano	hi-IN	lote de streaming	não	lote de streaming	batch (lote)	não	post-call
Húngaro	hu-HU	batch (lote)	não	batch (lote)	não	não	não
Islandês	is-IS	batch (lote)	não	batch (lote)	não	não	não

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Indonésio	id-ID	batch (lote)	não	batch (lote)	não	não	não
Italiano	it-IT	lote de streaming	não	lote de streaming	não	não	post-call, real-time
Japonês	ja-JP	lote de streaming	não	não	lote de streaming	não	post-call
Cabila	kab-DZ	batch (lote)	não	batch (lote)	não	não	não
Kannada	kn-IN	batch (lote)	não	batch (lote)	não	não	não
Cazaque	kk-KZ	batch (lote)	não	batch (lote)	não	não	não
Quiniana	rw-RW	batch (lote)	não	batch (lote)	não	não	não
Coreano	ko-KR	lote de streaming	não	não	não	não	post-call
Quirguiz	ky-KG	batch (lote)	não	batch (lote)	não	não	não
Letão	lv-LV	batch (lote)	não	batch (lote)	não	não	não
Lituano	lt-LT	batch (lote)	não	batch (lote)	não	não	não

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Luganda	lg-IN	batch (lote)	não	batch (lote)	não	não	não
Macedônio	mk-MK	batch (lote)	não	batch (lote)	não	não	não
Malaio	ms-MY	batch (lote)	não	batch (lote)	não	não	não
Malaiala	ml-IN	batch (lote)	não	batch (lote)	não	não	não
Maltês	mt-MT	batch (lote)	não	batch (lote)	não	não	não
Marathi	mr-IN	batch (lote)	não	batch (lote)	não	não	não
Mari das campinas	mhr-RU	batch (lote)	não	batch (lote)	não	não	não
Mongol	mn-MN	batch (lote)	não	batch (lote)	não	não	não
Norueguês Bokmål	no-NO	batch (lote)	não	batch (lote)	não	não	não
Oriá	or-IN	batch (lote)	não	batch (lote)	não	não	não
Pachto	ps-AF	batch (lote)	não	batch (lote)	não	não	não

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Polonês	pl-PL	batch (lote)	não	batch (lote)	não	não	não
Português	pt-PT	batch (lote)	não	batch (lote)	não	não	post-call
Português , brasileiro	pt-BR	lote de streaming	não	lote de streaming	não	não	post-call, real-time
Punjabi	pa-IN	batch (lote)	não	batch (lote)	não	não	não
Romeno	ro-RO	batch (lote)	não	batch (lote)	não	não	não
Russo	ru-RU	batch (lote)	não	não	não	não	não
Sérvio	sr-RS	batch (lote)	não	batch (lote)	não	não	não
Cingalês	si-LK	batch (lote)	não	batch (lote)	não	não	não
Eslovaco	sk-SK	batch (lote)	não	batch (lote)	não	não	não
Esloveno	sl-SI	batch (lote)	não	batch (lote)	não	não	não
Somali	so-SO	batch (lote)	não	batch (lote)	não	não	não

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Espanhol	es-ES	batch (lote)	não	batch (lote)	não	não	post-call
Espanhol, EUA	es-US	lote de streaming	não	lote de streaming	lote de streaming	lote de streaming	post-call, real-time
Sudanês	su-ID	batch (lote)	não	batch (lote)	não	não	não
Suaíli, Quênia	sw-KE	batch (lote)	não	batch (lote)	não	não	não
Suaíli, Burundi	sw-BI	batch (lote)	não	batch (lote)	não	não	não
Suaíli, Ruanda	sw-RW	batch (lote)	não	batch (lote)	não	não	não
Suaíli, Tanzânia	sw-TZ	batch (lote)	não	batch (lote)	não	não	não
Suaíli, Uganda	sw-UG	batch (lote)	não	batch (lote)	não	não	não
Sueco	sv-SE	batch (lote)	não	batch (lote)	não	não	não
Tagalo/filipino	tl-PH	batch (lote)	não	batch (lote)	não	não	não
Tâmil	ta-IN	batch (lote)	não	não	não	não	não

Idioma	Código do idioma	Entrada de dados	Transcrevendo números	Acrônimos	Modelos de idioma personalizados	Edição	Análise de chamadas*
Tatárico	tt-RU	batch (lote)	não	batch (lote)	não	não	não
Telugo	te-IN	batch (lote)	não	não	não	não	não
Tailandês	th-TH	lote de streaming	não	lote de streaming	não	não	não
Turco	tr-TR	batch (lote)	não	batch (lote)	não	não	não
Ucraniano	uk-UA	batch (lote)	não	batch (lote)	não	não	não
Uigur	ug-CN	batch (lote)	não	batch (lote)	não	não	não
Uzbeque	uz-UZ	batch (lote)	não	batch (lote)	não	não	não
Vietnamita	vi-VN	batch (lote)	não	batch (lote)	não	não	não
Galês	cy-WL	batch (lote)	não	batch (lote)	não	não	não
Uolofe	wo-SN	batch (lote)	não	batch (lote)	não	não	não
Zulu	zu-ZA	batch (lote)	não	batch (lote)	não	não	não

*Os seguintes insights do Call Analytics são oferecidos somente com alguns dialetos do inglês:

- [Resumo da chamada](#): en- * (todos os dialetos do inglês)
- [Detecção de problemas](#): en-AU, en-GB, en-US

Linguagens de programação compatíveis

Amazon Transcribe é compatível com os seguintes AWS SDKs:

Transcrições em lote	Transcrições de streaming
.NET	O .NET não é compatível com streaming.
AWS Interface de linha de comando (CLI)	A CLI não é compatível com streaming.
C++	C++
Go	Go
Java V2	Java V2
JavaScript	JavaScript V3
PHP V3	PHP V3
Python Boot3	SDK de streaming em Python para Amazon Transcribe
Ruby V3	Ruby V3
Rust	Rust

Para obter informações sobre como usar SDKs com Amazon Transcribe, consulte [Transcrevendo com os SDKs AWS](#)

Para obter mais informações sobre todos os AWS SDKs e ferramentas de criação disponíveis, consulte [Ferramentas para desenvolver](#). AWS


 Tip

Você pode encontrar exemplos de código do SDK nos seguintes GitHub repositórios:

- [AWS Exemplos de código](#)
- [Amazon Transcribe Exemplos](#)

Conjuntos de caracteres para vocabulários e filtros de vocabulário personalizados

Para cada idioma compatível com o Amazon Transcribe, há um conjunto específico de caracteres que o Amazon Transcribe pode reconhecer. Ao criar um vocabulário ou filtro de vocabulário personalizado, use somente os caracteres listados no conjunto de caracteres do seu idioma. Se você usar caracteres incompatíveis, o vocabulário personalizado ou filtro de vocabulário falhará.

 Important

Verifique se o arquivo de vocabulário personalizado usa somente os pontos de código Unicode compatíveis e as sequências de pontos de código listadas nos conjuntos de caracteres a seguir.

Muitos caracteres Unicode podem parecer idênticos em fontes populares, mesmo que usem pontos de código diferentes. Somente os pontos de código listados neste guia são compatíveis. Por exemplo, a palavra francesa déjà pode ser renderizada usando caracteres pré-compostos (em que um valor Unicode representa um caractere acentuado) ou caracteres decompostos (em que dois valores Unicode representam um caractere acentuado, um valor para o caractere base e outro para o acento).

- Versão pré-composta: 0064 **00E9** 006A **00E0** (renderiza como déjà)
- Versão decomposta: 0064 **0065 0301** 006A **0061 0300** (renderiza como déjà)

Tópicos

- [Conjunto de caracteres em abcázio](#)
- [Conjunto de caracteres em africâner](#)

- [Conjunto de caracteres árabes](#)
- [Conjunto de caracteres em asturiano](#)
- [Conjunto de caracteres em azerbaijano](#)
- [Conjunto de caracteres em armênio](#)
- [Conjunto de caracteres em basquir](#)
- [Conjunto de caracteres em basco](#)
- [Conjunto de caracteres em bielorrusso](#)
- [Conjunto de caracteres em bengali](#)
- [Conjunto de caracteres em bósnio](#)
- [Conjunto de caracteres em búlgaro](#)
- [Conjunto de caracteres em catalão](#)
- [Conjunto de caracteres em curdo central](#)
- [Conjunto de caracteres em chinês, mandarim \(China continental\), simplificado](#)
- [Conjunto de caracteres em chinês, mandarim \(Taiwan\), tradicional](#)
- [Conjunto de caracteres em croata](#)
- [Conjunto de caracteres em tcheco](#)
- [Conjunto de caracteres em dinamarquês](#)
- [Conjunto de caracteres em holandês](#)
- [Conjunto de caracteres em inglês](#)
- [Conjunto de caracteres em estoniano](#)
- [Conjunto de caracteres em farsi](#)
- [Conjunto de caracteres em finlandês](#)
- [Conjunto de caracteres em francês](#)
- [Conjunto de caracteres em galego](#)
- [Conjunto de caracteres em georgiano](#)
- [Conjunto de caracteres em alemão](#)
- [Conjunto de caracteres em grego](#)
- [Conjunto de caracteres em guzerate](#)
- [Conjunto de caracteres em hauçá](#)
- [Conjunto de caracteres em hebraico](#)

- [Conjunto de caracteres em hindi](#)
- [Conjunto de caracteres em húngaro](#)
- [Conjunto de caracteres em islandês](#)
- [Conjunto de caracteres em indonésio](#)
- [Conjunto de caracteres em italiano](#)
- [Conjunto de caracteres em japonês](#)
- [Conjunto de caracteres em cabila](#)
- [Conjunto de caracteres em canarês](#)
- [Conjunto de caracteres em cazaque](#)
- [Conjunto de caracteres em quiniaruanda](#)
- [Conjunto de caracteres em coreano](#)
- [Conjunto de caracteres em quirguiz](#)
- [Conjunto de caracteres em letão](#)
- [Conjunto de caracteres em lituano](#)
- [Conjunto de caracteres em luganda](#)
- [Conjunto de caracteres em macedônio](#)
- [Conjunto de caracteres em malaio](#)
- [Conjunto de caracteres em malaiala](#)
- [Conjunto de caracteres em maltês](#)
- [Conjunto de caracteres em marata](#)
- [Conjunto de caracteres em mari das campinas](#)
- [Conjunto de caracteres em mongol](#)
- [Conjunto de caracteres em bokmål norueguês](#)
- [Conjunto de caracteres em oria](#)
- [Conjunto de caracteres em pastó](#)
- [Conjunto de caracteres em polonês](#)
- [Conjunto de caracteres em português](#)
- [Conjunto de caracteres em punjabi](#)
- [Conjunto de caracteres em romeno](#)
- [Conjunto de caracteres em russo](#)

- [Conjunto de caracteres em sérvio](#)
- [Conjunto de caracteres em cingalês](#)
- [Conjunto de caracteres em eslovaco](#)
- [Conjunto de caracteres em esloveno](#)
- [Conjunto de caracteres em somali](#)
- [Conjunto de caracteres em espanhol](#)
- [Conjunto de caracteres em sundanês](#)
- [Conjunto de caracteres suaíli](#)
- [Conjunto de caracteres em sueco](#)
- [Conjunto de caracteres em tagalo/filipino](#)
- [Conjunto de caracteres em tâmil](#)
- [Conjunto de caracteres em tártaro](#)
- [Conjunto de caracteres em telugo](#)
- [Conjunto de caracteres em tailandês](#)
- [Conjunto de caracteres em turco](#)
- [Conjunto de caracteres em ucraniano](#)
- [Conjunto de caracteres em uigur](#)
- [Conjunto de caracteres em uzbeque](#)
- [Conjunto de caracteres em vietnamita](#)
- [Conjunto de caracteres em galês](#)
- [Conjunto de caracteres em uolofe](#)
- [Conjunto de caracteres em zulu](#)

Conjunto de caracteres em abcázio

Para vocabulários personalizados em abcázio, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
а	0430	лъ	0459
б	0431	њ	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	ѣ	0493
й	0439	ж	0497
к	043A	з	0499
л	043B	ќ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ћ	04A3
п	043F	н	04A5
р	0440	џ	04A9
с	0441	џ	04AB
т	0442	џ	04AD

Caractere	Código	Caractere	Código
у	0443	У	04AF
ф	0444	Ɔ	04B1
х	0445	Ი	04B3
ц	0446	Ლ	04B5
ч	0447	Ნ	04B7
ш	0448	Პ	04BB
щ	0449	Რ	04BD
ъ	044A	Ტ	04BF
ы	044B	Ფ	04CA
ь	044C	Ღ	04D1
э	044D	Შ	04D3
ю	044E	Ც	04D7
я	044F	Წ	04D9
#	0450	Ხ	04E1
ë	0451	Ჰ	04E3
ђ	0452	Ჲ	04E7
í	0453	Ჴ	04E9
є	0454	Ჶ	04EF
s	0455	Ჸ	04F1
i	0456	Ჺ	04F3

Caractere	Código	Caractere	Código
ï	0457	#	04F7
j	0458	ÿ	04F9
#	0525		

Conjunto de caracteres em africâner

Para vocabulários personalizados em africâner, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
á	00E1	ï	00EF
è	00E8	ó	00F3
é	00E9	ô	00F4
ê	00EA	ö	00F6
ë	00EB	ú	00FA
í	00ED	û	00FB
î	00EE	ü	00FC

Conjunto de caracteres árabes

Para vocabulários personalizados em árabe, é possível usar os seguintes caracteres Unicode no campo `Phrase`. Também é possível usar o caractere hífen (-) para separar palavras.

Caractere	Código	Caractere	Código
ء	0621	س	0633
آ	0622	ش	0634
أ	0623	ص	0635
ؤ	0624	ض	0636
إ	0625	ط	0637
ئ	0626	ظ	0638
ا	0627	ع	0639
ب	0628	غ	063A
ة	0629	ف	0641
ت	062A	ق	0642
ث	062B	ك	0643
ج	062C	ل	0644
ح	062D	م	0645
خ	062E	ن	0646
د	062F	ه	0647
ذ	0630	و	0648
ر	0631	ى	0649
ز	0632	ي	064A

Conjunto de caracteres em asturiano

Para vocabulários personalizados em asturiano, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
á	00E1	ñ	00F1
é	00E9	ó	00F3
í	00ED	ú	00FA
ü	00FC		

Conjunto de caracteres em azerbaijano

Para vocabulários personalizados em azerbaijano, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ä	00E4	ğ	011F

Caractere	Código	Caractere	Código
ç	00E7	ı	0131
ö	00F6	ş	015F
ü	00FC	ə	0259
.	0307		

Conjunto de caracteres em armênio

Para vocabulários personalizados em armênio, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ա	0561	ւ	0574
բ	0562	յ	0575
գ	0563	ի	0576
դ	0564	ե	0577
ե	0565	ն	0578
զ	0566	շ	0579
է	0567	ղ	057A
ը	0568	ջ	057B

Caractere	Código	Caractere	Código
р	0569	и	057C
ґ	056A	и	057D
һ	056B	ҫ	057E
л	056C	и	057F
һ	056D	р	0580
ґ	056E	g	0581
һ	056F	л	0582
һ	0570	һ	0583
а	0571	f	0584
һ	0572	o	0585
һ	0573	һ	0586

Conjunto de caracteres em basquir

Para vocabulários personalizados em basquir, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
a	0430	ӡ	0459

Caractere	Código	Caractere	Código
б	0431	һ	045A
в	0432	һ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ӳ	045E
ж	0436	ц	045F
з	0437	г	0491
и	0438	Ғ	0493
й	0439	җ	0497
к	043A	ҙ	0499
л	043B	қ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ң	04A3
п	043F	п	04A5
р	0440	ә	04A9
с	0441	с	04AB
т	0442	т	04AD
у	0443	у	04AF
ф	0444	ф	04B1

Caractere	Código	Caractere	Código
х	0445	χ	04B3
ц	0446	ц	04B5
ч	0447	ч	04B7
ш	0448	h	04BB
щ	0449	є	04BD
ъ	044A	є	04BF
ы	044B	#	04CA
ь	044C	ă	04D1
э	044D	ă	04D3
ю	044E	ě	04D7
я	044F	ə	04D9
#	0450	з	04E1
ë	0451	й	04E3
ђ	0452	ö	04E7
í	0453	ө	04E9
є	0454	ÿ	04EF
s	0455	ÿ	04F1
i	0456	ÿ	04F3
ï	0457	#	04F7
j	0458	Ы	04F9

Conjunto de caracteres em basco

Para vocabulários personalizados em basco, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
á	00E1	ñ	00F1
é	00E9	ó	00F3
í	00ED	ú	00FA
ü	00FC		

Conjunto de caracteres em bielorrusso

Para vocabulários personalizados em bielorrusso, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
a	0430	c	0441

Caractere	Código	Caractere	Código
б	0431	т	0442
в	0432	у	0443
г	0433	ф	0444
д	0434	х	0445
е	0435	ц	0446
ж	0436	ч	0447
з	0437	ш	0448
й	0439	ы	044B
к	043A	ь	044C
л	043B	э	044D
м	043C	ю	044E
н	043D	я	044F
о	043E	ë	0451
п	043F	і	0456
р	0440	ŷ	045E

Conjunto de caracteres em bengali

Para vocabulários personalizados em bengali, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
ূ	0981	দ	09A6
ৃ	0982	ধ	09A7
ৄ	0983	ন	09A8
অ	0985	প	09AA
আ	0986	ফ	09AB
ঈ	0987	ব	09AC
ঐ	0988	ভ	09AD
ঊ	0989	ম	09AE
ঋ	098A	য	09AF
ঌ	098B	র	09B0
এ	098F	ল	09B2
ঐ	0990	শ	09B6
ও	0993	ষ	09B7
ঔ	0994	স	09B8
ক	0995	হ	09B9
খ	0996	.	09BC
গ	0997	#	09BD
ঘ	0998	†	09BE
ঙ	0999	†	09BF

Caractere	Código	Caractere	Código
চ	099A	ঁ	09C0
ছ	099B	্	09C1
জ	099C	্	09C2
ঝ	099D	্	09C3
ঞ	099E	্	09C4
ট	099F	্	09C7
ঠ	09A0	্	09C8
ড	09A1	্	09CB
ঢ	09A2	্	09CC
ণ	09A3	্	09CD
ত	09A4	#	09CE
থ	09A5	ঁ	09D7

Conjunto de caracteres em bósnio

Para vocabulários personalizados em bósnio, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ć	0107	đ	0111
č	010D	š	0161
ž	017E		

Conjunto de caracteres em búlgaro

Para vocabulários personalizados em búlgaro, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
а	0430	п	043F
б	0431	р	0440
в	0432	с	0441
г	0433	т	0442
д	0434	у	0443
е	0435	ф	0444
ж	0436	х	0445
з	0437	ц	0446
и	0438	ч	0447

Caractere	Código	Caractere	Código
й	0439	ш	0448
к	043A	щ	0449
л	043B	ъ	044A
м	043C	ь	044C
н	043D	ю	044E
о	043E	я	044F

Conjunto de caracteres em catalão

Para vocabulários personalizados em catalão, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
à	00E0	ï	00EF
ç	00E7	ò	00F2
è	00E8	ó	00F3
é	00E9	ú	00FA
í	00ED	ü	00FC
ı	0140		

Conjunto de caracteres em curdo central

Para vocabulários personalizados em curso central, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ئ	0626	م	0645
ا	0627	ن	0646
ب	0628	و	0648
ت	062A	پ	067E
ج	062C	چ	0686
ح	062D	ر	0695
خ	062E	ژ	0698
د	062F	ڤ	06A4
ر	0631	ک	06A9
ز	0632	گ	06AF
س	0633	ڭ	06B5
ش	0634	ه	06BE
ع	0639	ۆ	06C6
غ	063A	ؤ	06C7

Caractere	Código	Caractere	Código
ف	0641	س	06CC
ق	0642	ئ	06CE
ج	0644	ه	06D5

Conjunto de caracteres em chinês, mandarim (China continental), simplificado

Para vocabulários personalizados em chinês (simplificado), o campo `Phrase` pode usar qualquer um dos caracteres listados no seguinte arquivo:

- [zh-cn-character-set](#)

O campo `SoundsLike` pode conter as sílabas pinyin listadas no seguinte arquivo:

- [pinyin-character-set](#)

Quando você usar sílabas pinyin no campo `SoundsLike`, separe-as com um hífen (-).

O Amazon Transcribe representa os quatro tons em chinês (simplificado) usando números. A tabela a seguir mostra como as marcas de tom são mapeadas para a palavra “ma”.

Tom	Marca de tom	Número do tom
Tom 1	mā	ma1
Tom 2	má	ma2
Tom 3	mǎ	ma3
Tom 4	mà	ma4

Note

Para o 5º tom (neutro), você pode usar o Tom 1, com exceção de “er”, que deve ser mapeado para o Tom 2. Por exemplo, 打转儿 seria representado como “da3-zhuan4-er2”.

Os vocabulários personalizados em chinês (simplificado) não usam o campo IPA, mas você ainda deve incluir o cabeçalho IPA na tabela de vocabulário personalizado.

Veja a seguir um exemplo de arquivo de entrada em formato de texto. O exemplo usa espaços para alinhar as colunas. Seus arquivos de entrada devem usar caracteres TAB para separar as colunas. Inclua espaços apenas na coluna DisplayAs.

Phrase	SoundsLike	IPA	DisplayAs
##	kang1-jian4		
##	qian3-ze2		
####	guo2-fang2-da4-chen2		
#####	shi4-jie4-bo2-lan3-hui4		###

Conjunto de caracteres em chinês, mandarim (Taiwan), tradicional

Para vocabulários personalizados em chinês (tradicional), o campo Phrase pode usar qualquer um dos caracteres listados no seguinte arquivo:

- [zh-tw-character-set](#)

O campo SoundsLike pode conter as sílabas zhuyin listadas no seguinte arquivo:

- [zhuyin-character-set](#)

Quando você usar sílabas zhuyin no campo SoundsLike, separe-as com um hífen (-).

O Amazon Transcribe representa os quatro tons em chinês (tradicional) usando números. A tabela a seguir mostra como as marcas de tom são mapeadas para a palavra ㄇㄩˊ.

Tom	Marca de tom	
Tom 1	ㄇㄩˊ	

Tom	Marca de tom	
Tom 2	□Υ´	
Tom 3	□Υ˘	
Tom 4	□Υ`	

Os vocabulários personalizados em chinês (tradicional) não usam o campo IPA, mas você ainda deve incluir o cabeçalho IPA na tabela de vocabulário personalizado.

Veja a seguir um exemplo de arquivo de entrada em formato de texto. O exemplo usa espaços para alinhar as colunas. Seus arquivos de entrada devem usar caracteres TAB para separar as colunas. Inclua espaços apenas na coluna `DisplayAs`.

Phrase	SoundsLike	IPA	DisplayAs
##	###`-##		
##	###˘-##´		
####	###´-##´-##`-##˘		
#####	#`-###`-##´-##˘-###`	###	

Conjunto de caracteres em croata

Para vocabulários personalizados em croata, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ć	0107	đ	0111
č	010D	š	0161

Caractere	Código	Caractere	Código
ž	017E		

Conjunto de caracteres em tcheco

Para vocabulários personalizados em tcheco, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
á	00E1	ď	010F
é	00E9	ě	011B
í	00ED	ň	0148
ó	00F3	ř	0159
ú	00FA	š	0161
ý	00FD	ť	0165
č	010D	ů	016F
ž	017E		

Conjunto de caracteres em dinamarquês

Para vocabulários personalizados em dinamarquês, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
Å	00C5	æ	00E6
Æ	00C6	é	00E9
Ø	00D8	ø	00F8
å	00E5		

Conjunto de caracteres em holandês

Para vocabulários personalizados em holandês, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
à	00E0	î	00EE
á	00E1	ï	00EF

Caractere	Código	Caractere	Código
â	00E2	ñ	00F1
ä	00E4	ò	00F2
ç	00E7	ó	00F3
è	00E8	ô	00F4
é	00E9	ö	00F6
ê	00EA	ù	00F9
ë	00EB	ú	00FA
ì	00EC	û	00FB
í	00ED	ü	00FC

Conjunto de caracteres em inglês

Para vocabulários personalizados em inglês, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Conjunto de caracteres em estoniano

Para vocabulários personalizados em estoniano, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)

- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
ä	00E4	ü	00FC
õ	00F5	š	0161
ö	00F6	ž	017E

Conjunto de caracteres em farsi

Para vocabulários personalizados em farsi, é possível usar os seguintes caracteres no campo Phrase.

Caractere	Código	Caractere	Código
ء	0621	ظ	0638
آ	0622	ع	0639
أ	0623	غ	063A
ؤ	0624	ف	0641
ئ	0626	ق	0642
ا	0627	ج	0644
ب	0628	م	0645
ت	062A	ن	0646
ث	062B	ه	0647
ح	062C	و	0648

Caractere	Código	Caractere	Código
ح	062D	ـ	064E
خ	062E	ء	064F
د	062F	ـ	0650
ذ	0630	ء	0651
ر	0631	پ	067E
ز	0632	چ	0686
س	0633	ژ	0698
ش	0634	ک	06A9
ص	0635	گ	06AF
ض	0636	ی	06CC
ط	0637		

Conjunto de caracteres em finlandês

Para vocabulários personalizados em finlandês, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
ä	00E4	ö	00F6

Caractere	Código	Caractere	Código
å	00E5	š	0161
ž	017E		

Conjunto de caracteres em francês

Para vocabulários personalizados em francês, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
À	00C0	à	00E0
Â	00C2	â	00E2
Ç	00C7	ç	00E7
È	00C8	è	00E8
É	00C9	é	00E9
Ê	00CA	ê	00EA
Ë	00CB	ë	00EB
Î	00CE	î	00EE
Ï	00CF	ï	00EF

Caractere	Código	Caractere	Código
Ô	00D4	ô	00F4
Ö	00D6	ö	00F6
Ù	00D9	ù	00F9
Û	00DB	û	00FB
Ü	00DC	ü	00FC

Conjunto de caracteres em galego

Para vocabulários personalizados em galego, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
á	00E1	ñ	00F1
é	00E9	ó	00F3
í	00ED	ú	00FA
ü	00FC		

Conjunto de caracteres em georgiano

Para vocabulários personalizados em georgiano, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
ა	10D0	ბ	10E0
ბ	10D1	ს	10E1
გ	10D2	ძ	10E2
დ	10D3	წ	10E3
ე	10D4	ჭ	10E4
ვ	10D5	ყ	10E5
ზ	10D6	ღ	10E6
თ	10D7	ყ	10E7
ო	10D8	შ	10E8
პ	10D9	ჩ	10E9
რ	10DA	ც	10EA
ს	10DB	ძ	10EB
ტ	10DC	წ	10EC
კ	10DD	ჭ	10ED
ხ	10DE	ხ	10EE
ჯ	10DF	ჯ	10EF

Caractere	Código	Caractere	Código
ÿ	10F0		

Conjunto de caracteres em alemão

Para vocabulários personalizados em alemão, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ä	00E4	Ä	00C4
ö	00F6	Ö	00D6
ü	00FC	Ü	00DC
ß	00DF		

Conjunto de caracteres em grego

Para vocabulários personalizados em grego, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)

- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
ά	03AC	ν	03BD
έ	03AD	ξ	03BE
ή	03AE	ο	03BF
ί	03AF	π	03C0
ü	03B0	ρ	03C1
α	03B1	ς	03C2
β	03B2	σ	03C3
γ	03B3	τ	03C4
δ	03B4	υ	03C5
ε	03B5	φ	03C6
ζ	03B6	χ	03C7
η	03B7	ψ	03C8
θ	03B8	ω	03C9
ι	03B9	ï	03CA
κ	03BA	ü	03CB
λ	03BB	ó	03CC
μ	03BC	ώ	03CE
ĩ	0390		

Conjunto de caracteres em guzerate

Para vocabulários personalizados em guzerate, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ઁ	0A81	ઃ	0AA6
ઃ	0A82	ઘ	0AA7
ઃ	0A83	ઙ	0AA8
અ	0A85	પ	0AAA
આ	0A86	ફ	0AAB
ઇ	0A87	બ	0AAC
ઈ	0A88	ભ	0AAD
ઉ	0A89	મ	0AAE
ઊ	0A8A	ય	0AAF
ઋ	0A8B	ર	0AB0
ઌ	0A8D	લ	0AB2
એ	0A8F	ળ	0AB3
ઐ	0A90	વ	0AB5
ઑ	0A91	શ	0AB6

Caractere	Código	Caractere	Código
ઓ	0A93	બ	0AB7
ઔ	0A94	સ	0AB8
ક	0A95	હ	0AB9
ખ	0A96	.	0ABC
ગ	0A97	૧	0ABE
ઘ	0A98	૨	0ABF
ઙ	0A99	૩	0AC0
ચ	0A9A	૪	0AC1
છ	0A9B	૫	0AC2
જ	0A9C	૬	0AC3
ઝ	0A9D	૭	0AC5
ઞ	0A9E	૮	0AC7
ટ	0A9F	૯	0AC8
ઠ	0AA0	૧૦	0AC9
ડ	0AA1	૧૧	0ACB
ઢ	0AA2	૧૨	0ACC
ણ	0AA3	૧૩	0ACD
ત	0AA4	૩૦	0AD0
થ	0AA5	૩૧	0AE0

Conjunto de caracteres em hauçá

Para vocabulários personalizados em hauçá, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
k	0199	ḅ	0253
y	01B4	ḍ	0257
~	0303		

Conjunto de caracteres em hebraico

Para vocabulários personalizados em hebraico, é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
-	002D	◌׀	05DD
κ	05D0	◌ׁ	05DE
כ	05D1	◌ׂ	05DF
λ	05D2	◌׃	05E0
τ	05D3	◌ׄ	05E1
η	05D4	◌ׅ	05E2

Caractere	Código	Caractere	Código
ॠ	05D5	ॡ	05E3
ॢ	05D6	ॣ	05E4
।	05D7	॥	05E5
०	05D8	०	05E6
ॡ	05D9	ॡ	05E7
ॢ	05DA	ॢ	05E8
ॣ	05DB	ॣ	05E9
।	05DC	।	05EA

Conjunto de caracteres em hindi

Para vocabulários personalizados em hindi, você pode usar os seguintes caracteres no campo `Phrase`:

Caractere	Código	Caractere	Código
-	002D	थ	0925
.	002E	द	0926
ॆ	0901	ध	0927
.	0902	न	0928
:	0903	प	092A
अ	0905	फ	092B
आ	0906	ब	092C
इ	0907	भ	092D

Caractere	Código	Caractere	Código
ई	0908	म	092E
उ	0909	य	092F
ऊ	090A	र	0930
ऋ	090B	ल	0932
ए	090F	व	0935
ऐ	0910	श	0936
ऑ	0911	ष	0937
ओ	0913	स	0938
औ	0914	ह	0939
क	0915	ट	093E
ख	0916	ठि	093F
ग	0917	ति	0940
घ	0918	ु	0941
ङ	0919	ॆ	0942
च	091A	ॆ	0943
छ	091B	ॆ	0945
ज	091C	ॆ	0947
झ	091D	ॆ	0948
ञ	091E	ँ	0949
ट	091F	ॆ	094B

Caractere	Código	Caractere	Código
ठ	0920	तै	094C
ड	0921	ँ	094D
ढ	0922	ज़	095B
ण	0923	ड़	095C
त्	0924	ढ	095D

O Amazon Transcribe mapeia os seguintes caracteres:

Caractere	Mapeado para
न (0929)	न (0928)
र (0931)	र (0930)
क (0958)	क (0915)
ख (0959)	ख (0916)
ग (095A)	ग (0917)
फ (095E)	फ (092B)
य (095F)	य (092F)

Conjunto de caracteres em húngaro

Para vocabulários personalizados em húngaro, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
á	00E1	ö	00F6
é	00E9	ú	00FA
í	00ED	ü	00FC
ó	00F3	õ	0151
ű	0171		

Conjunto de caracteres em islandês

Para vocabulários personalizados em islandês, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
á	00E1	ú	00FA
é	00E9	ý	00FD
ð	00F0	þ	00FE
í	00ED	æ	00E6
ó	00F3	ö	00F6

Conjunto de caracteres em indonésio

Para vocabulários personalizados em indonésio, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Conjunto de caracteres em italiano

Para vocabulários personalizados em italiano, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
À	00C0	à	00E0
Ä	00C4	ä	00E4
Ç	00C7	ç	00E7
È	00C8	è	00E8
É	00C9	é	00E9

Caractere	Código	Caractere	Código
Ê	00CA	ê	00EA
Ë	00CB	ë	00EB
Ì	00CC	ì	00EC
Ò	00D2	ò	00F2
Ù	00D9	ù	00F9
Ü	00DC	ü	00FC

Conjunto de caracteres em japonês

Para vocabulários personalizados em japonês, o campo `DisplayAs` é compatível com todos os caracteres hiragana, katakana e kanji e letras maiúsculas romaji de largura total.

O campo `Phrase` é compatível com os caracteres listados no seguinte arquivo:

- [ja-jp-character-set](#)

Conjunto de caracteres em cabila

Para vocabulários personalizados em cabila, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ï	00EF	ḍ	1E0D

Caractere	Código	Caractere	Código
č	010D	ħ	1E25
ř	0159	ŗ	1E5B
ǧ	01E7	ş	1E63
ε	025B	ţ	1E6D
γ	0263	ẓ	1E93

Conjunto de caracteres em canarês

Para vocabulários personalizados em canarês, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
೦	0C82	೧	0CA7
೨	0C83	೨	0CA8
೩	0C85	೩	0CAA
೪	0C86	೪	0CAB
೫	0C87	೫	0CAC
೬	0C88	೬	0CAD
೭	0C89	೭	0CAE

Caractere	Código	Caractere	Código
ಊ	0C8A	ಯ	0CAF
ಋ	0C8B	ರ	0CB0
ಎ	0C8E	ಲ	0CB2
ಏ	0C8F	ಳ	0CB3
ಐ	0C90	ಱ	0CB5
ಒ	0C92	ಠ	0CB6
ಓ	0C93	ಱ	0CB7
ಔ	0C94	ಱ	0CB8
ಕ	0C95	ಱ	0CB9
ಖ	0C96	#	0CBC
ಗ	0C97	#	0CBD
ಘ	0C98	ಁ	0CBE
ಙ	0C99	ಂ	0CBF
ಚ	0C9A	ಃ	0CC0
ಛ	0C9B	಄	0CC1
ಜ	0C9C	ಅ	0CC2
ಝ	0C9D	ಆ	0CC3
ಞ	0C9E	ಇ	0CC6
ಟ	0C9F	ಈ	0CC7
ಠ	0CA0	ಉ	0CC8

Caractere	Código	Caractere	Código
Ә	0CA1	Ә	0CCA
Ғ	0CA2	Ғ	0CCB
Ң	0CA3	Ң	0CCC
Ң	0CA4	Ң	0CCD
Ғ	0CA5	Ғ	0CD5
Ғ	0CA6	Ғ	0CD6
Ә	0CE0		

Conjunto de caracteres em cazaque

Para vocabulários personalizados em cazaque, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
т	0442	ы	044B
б	0431	я	044F
о	043E	с	0441
п	043F	h	04BB
ш	0448	д	0434

Caractere	Código	Caractere	Código
и	0438	р	0440
ч	0447	г	0433
н	043D	ё	0451
қ	049B	й	0439
і	0456	ө	04E9
щ	0449	в	0432
е	0435	э	044D
ә	04D9	ң	04A3
ю	044E	л	043B
з	0437	ф	0444
х	0445	к	043A
ц	0446	у	0443
ү	04AF	ж	0436
м	043C	ғ	0493
ь	044C	а	0430
ъ	044A	ұ	04B1

Conjunto de caracteres em quiniaruanda

Para vocabulários personalizados em quiniaruanda, você pode usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)

- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
á	00E1	ó	00F3
â	00E2	ô	00F4
ã	00E3	ú	00FA
ç	00E7	ü	00FC
è	00E8	ā	0101
é	00E9	ē	0113
ê	00EA	ī	012B
ë	00EB	ō	014D
í	00ED	ū	016B
ï	00EF	’	0301

Conjunto de caracteres em coreano

Para vocabulários personalizados em coreano, você pode usar qualquer uma das sílabas Hangul no campo Phrase. Para obter mais informações, consulte [sílabas Hangul](#) na Wikipédia.

Conjunto de caracteres em quirguiz

Para vocabulários personalizados em quirguiz, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
а	0430	лъ	0459
б	0431	њ	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	ѣ	0493
й	0439	ж	0497
к	043A	з	0499
л	043B	ќ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ћ	04A3
п	043F	н	04A5
р	0440	џ	04A9
с	0441	џ	04AB
т	0442	џ	04AD

Caractere	Código	Caractere	Código
у	0443	У	04AF
ф	0444	Ф	04B1
х	0445	Х	04B3
ц	0446	Ц	04B5
ч	0447	Ч	04B7
ш	0448	Ш	04BB
щ	0449	Щ	04BD
ъ	044A	Ъ	04BF
ы	044B	Ы	04CA
ь	044C	Ь	04D1
э	044D	Э	04D3
ю	044E	Ю	04D7
я	044F	Я	04D9
#	0450	З	04E1
ë	0451	Й	04E3
ђ	0452	Ö	04E7
í	0453	Ө	04E9
є	0454	ȳ	04EF
s	0455	ÿ	04F1
i	0456	ŷ	04F3

Caractere	Código	Caractere	Código
ï	0457	#	04F7
j	0458	ÿ	04F9

Conjunto de caracteres em letão

Para vocabulários personalizados em letão, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ā	0101	ķ	0137
č	010D	ļ	013C
ē	0113	ņ	0146
ģ	0123	š	0161
ī	012B	ū	016B
ž	017E		

Conjunto de caracteres em lituano

Para vocabulários personalizados em lituano, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z

- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ą	0105	ı	012F
č	010D	š	0161
ę	0119	ı	0173
è	0117	ū	016B
ž	017E		

Conjunto de caracteres em luganda

Para vocabulários personalizados em luganda, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ÿ	00FF	η	014B

Conjunto de caracteres em macedônio

Para vocabulários personalizados em macedônio, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
а	0430	лъ	0459
б	0431	њ	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	ѣ	0493
й	0439	ж	0497
к	043A	ѕ	0499
л	043B	ќ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ћ	04A3
п	043F	н	04A5

Caractere	Código	Caractere	Código
ρ	0440	ϱ	04A9
с	0441	ç	04AB
т	0442	ṭ	04AD
у	0443	γ	04AF
φ	0444	ϣ	04B1
х	0445	χ	04B3
ц	0446	ц	04B5
ч	0447	ч	04B7
ш	0448	h	04BB
щ	0449	є	04BD
ъ	044A	є	04BF
ы	044B	#	04CA
ь	044C	ă	04D1
э	044D	ä	04D3
ю	044E	ě	04D7
я	044F	ə	04D9
#	0450	з	04E1
ë	0451	ñ	04E3
ђ	0452	ö	04E7
ѓ	0453	ө	04E9

Caractere	Código	Caractere	Código
ε	0454	ȳ	04EF
s	0455	ÿ	04F1
i	0456	ÿ	04F3
ï	0457	#	04F7
j	0458	ÿ	04F9

Conjunto de caracteres em malaio

Para vocabulários personalizados em malaio, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Conjunto de caracteres em malaiala

Para vocabulários personalizados em malaiala, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
o	0D02	ന	0D28
ഃ	0D03	പ	0D2A
അ	0D05	ഫ	0D2B
ആ	0D06	ബ	0D2C
ഇ	0D07	ഭ	0D2D
ഈ	0D08	മ	0D2E
ഉ	0D09	യ	0D2F
ഊ	0D0A	ര	0D30
ഋ	0D0B	ഠ	0D31
എ	0D0E	ല	0D32
ഏ	0D0F	ള	0D33
ഐ	0D10	ഴ	0D34
ഒ	0D12	വ	0D35
ഓ	0D13	ശ	0D36
ഔ	0D14	ഷ	0D37
ക	0D15	സ	0D38
ഖ	0D16	ഹ	0D39
ഗ	0D17	ഘ	0D3E
ഘ	0D18	ങ	0D3F
ങ	0D19	ച	0D40

Caractere	Código	Caractere	Código
ᐅ	0D1A	ᐆ	0D41
ᐇ	0D1B	ᐈ	0D42
ᐉ	0D1C	ᐊ	0D43
ᐋ	0D1D	ᐌ	0D46
ᐍ	0D1E	ᐎ	0D47
ᐏ	0D1F	ᐐ	0D48
ᐑ	0D20	ᐒ	0D4A
ᐓ	0D21	ᐔ	0D4B
ᐕ	0D22	ᐖ	0D4C
ᐗ	0D23	ᐘ	0D4D
ᐙ	0D24	#	0D7A
ᐛ	0D25	#	0D7B
ᐝ	0D26	#	0D7C
ᐟ	0D27	#	0D7D

Conjunto de caracteres em maltês

Para vocabulários personalizados em maltês, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
à	00E0	ù	00F9
è	00E8	ć	010B
ì	00EC	ġ	0121
ò	00F2	ħ	0127
ž	017C		

Conjunto de caracteres em marata

Para vocabulários personalizados em marata, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ॆ	0901	थ	0925
ॆ	0902	द	0926
:	0903	ध	0927
अ	0905	न	0928
आ	0906	प	092A
इ	0907	फ	092B

Caractere	Código	Caractere	Código
ई	0908	ब	092C
उ	0909	भ	092D
ऊ	090A	म	092E
ऋ	090B	य	092F
ॠ	090D	र	0930
ए	090F	ल	0932
ऐ	0910	ळ	0933
ऑ	0911	व	0935
ओ	0913	श	0936
औ	0914	ष	0937
क	0915	स	0938
ख	0916	ह	0939
ग	0917	.	093C
घ	0918	।	093E
ङ	0919	ि	093F
च	091A	ी	0940
छ	091B	ु	0941
ज	091C	ू	0942
झ	091D	े	0943
ञ	091E	ै	0945

Caractere	Código	Caractere	Código
ट	091F	˘	0947
ठ	0920	˙	0948
ड	0921	ँ	0949
ढ	0922	ं	094B
ण	0923	ँ	094C
त	0924	˘	094D
ॐ	0950		

Conjunto de caracteres em mari das campinas

Para vocabulários personalizados em mari das campinas, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
а	0430	љ	0459
б	0431	њ	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D

Caractere	Código	Caractere	Código
е	0435	ÿ	045E
ж	0436	ц	045F
з	0437	г	0491
и	0438	ф	0493
й	0439	ж	0497
к	043A	з	0499
л	043B	қ	049B
м	043C	к	049F
н	043D	т	04A1
о	043E	ң	04A3
п	043F	н	04A5
р	0440	œ	04A9
с	0441	ç	04AB
т	0442	т	04AD
у	0443	ү	04AF
ф	0444	ұ	04B1
х	0445	х	04B3
ц	0446	ц	04B5
ч	0447	ч	04B7
ш	0448	h	04BB

Caractere	Código	Caractere	Código
щ	0449	е	04BD
ъ	044A	ӗ	04BF
ы	044B	#	04CA
ь	044C	ă	04D1
э	044D	ä	04D3
ю	044E	ě	04D7
я	044F	ə	04D9
#	0450	з	04E1
ë	0451	й	04E3
ђ	0452	ö	04E7
ѓ	0453	ө	04E9
є	0454	ÿ	04EF
s	0455	ÿ̇	04F1
i	0456	ÿ̈	04F3
ï	0457	#	04F7
j	0458	Ы	04F9

Conjunto de caracteres em mongol

Para vocabulários personalizados em mongol, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)

- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
а	0430	ль	0459
б	0431	њ	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	Ѡ	0493
й	0439	ѡ	0497
к	043A	Ѣ	0499
л	043B	ѣ	049B
м	043C	Ѥ	049F
н	043D	Ѧ	04A1
о	043E	ѧ	04A3
п	043F	Ѩ	04A5
р	0440	Ѱ	04A9
с	0441	ѱ	04AB

Caractere	Código	Caractere	Código
т	0442	ᠮ	04AD
у	0443	ᠮ	04AF
ф	0444	ᠮ	04B1
х	0445	ᠮ	04B3
ц	0446	ᠮ	04B5
ч	0447	ᠮ	04B7
ш	0448	h	04BB
щ	0449	ᠮ	04BD
ъ	044A	ᠮ	04BF
ы	044B	#	04CA
ь	044C	ᠮ	04D1
э	044D	ᠮ	04D3
ю	044E	ᠮ	04D7
я	044F	ᠮ	04D9
#	0450	з	04E1
ë	0451	ᠮ	04E3
ñ	0452	ö	04E7
í	0453	ᠮ	04E9
є	0454	ᠮ	04EF
s	0455	ᠮ	04F1

Caractere	Código	Caractere	Código
i	0456	ý	04F3
ï	0457	#	04F7
j	0458	ÿ	04F9

Conjunto de caracteres em bokmål norueguês

Para vocabulários personalizados em bokmål, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
å	00E5	æ	00E6
ø	00F8		

Conjunto de caracteres em oriá

Para vocabulários personalizados em oriá, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
◌	0B01	୩	0B26
◌	0B02	୪	0B27
୫	0B03	୬	0B28
୭	0B05	୮	0B2A
୯	0B06	ୱ	0B2B
ୱ	0B07	ୱ	0B2C
ୱ	0B08	ୱ	0B2D
ୱ	0B09	ୱ	0B2E
ୱ	0B0A	ୱ	0B2F
ୱ	0B0B	ୱ	0B30
ୱ	0B0F	ୱ	0B32
ୱ	0B10	ୱ	0B33
ୱ	0B13	ୱ	0B36
ୱ	0B14	ୱ	0B37
ୱ	0B15	ୱ	0B38
ୱ	0B16	ୱ	0B39
ୱ	0B17	.	0B3C
ୱ	0B18		0B3E
ୱ	0B19	ˆ	0B3F
ୱ	0B1A	1	0B40

Caractere	Código	Caractere	Código
ځ	0B1B	ا	0B41
ښ	0B1C	ب	0B42
د	0B1D	پ	0B43
څ	0B1E	څ	0B47
ډ	0B1F	ډ	0B48
و	0B20	و#	0B4B
ږ	0B21	ږ	0B4C
ښ	0B22	ـ	0B4D
ښ	0B23	ـ	0B56
ښ	0B24	ښ	0B5F
ښ	0B25	ښ	0B60
#	0B71		

Conjunto de caracteres em pastó

Para vocabulários personalizados em pashto, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
آ	0622	و	0648
أ	0623	ي	064A
ؤ	0624	=	064B
ئ	0626	ة	064C
ا	0627	=	064D
ب	0628	ـ	064E
ت	062A	ة	064F
ث	062B	ـ	0650
ج	062C	ة	0651
ح	062D	ة	0652
خ	062E	#	0654
د	062F	ـ	0670
ذ	0630	ذ	067C
ر	0631	ر	067E
ز	0632	ز	0681
س	0633	س	0685
ش	0634	ش	0686
ص	0635	ص	0689
ض	0636	ض	0693
ط	0637	ط	0696

Caractere	Código	Caractere	Código
ظ	0638	ژ	0698
ع	0639	بن	069A
غ	063A	ک	06A9
ف	0641	گ	06AB
ق	0642	گ	06AF
ل	0644	ن	06BC
م	0645	ی	06CC
ن	0646	ی	06CD
ه	0647	ې	06D0

Conjunto de caracteres em polonês

Para vocabulários personalizados em polonês, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
ó	00F3	ł	0142
ą	0105	ń	0144
ć	0107	ś	015B

Caractere	Código	Caractere	Código
ę	0119	ź	017A
ż	017C		

Conjunto de caracteres em português

Para vocabulários personalizados em português, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
À	00C0	à	00E0
Á	00C1	á	00E1
Â	00C2	â	00E2
Ã	00C3	ã	00E3
Ä	00C4	ä	00E4
Ç	00C7	ç	00E7
È	00C8	è	00E8
É	00C9	é	00E9
Ê	00CA	ê	00EA

Caractere	Código	Caractere	Código
Ë	00CB	ë	00EB
Í	00CD	í	00ED
Ñ	00D1	ñ	00F1
Ó	00D3	ó	00F3
Ô	00D4	ô	00F4
Õ	00D5	õ	00F5
Ö	00D6	ö	00F6
Ú	00DA	ú	00FA
Ü	00DC	ü	00FC

Conjunto de caracteres em punjabi

Para vocabulários personalizados em punjabi, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
ਅ	0A05	ੳ	0A27
ਆ	0A06	ਠ	0A28
ੲ	0A07	ਘ	0A2A

Caractere	Código	Caractere	Código
ਈ	0A08	ਫ	0A2B
ਉ	0A09	ਬ	0A2C
ਊ	0A0A	ਭ	0A2D
ਏ	0A0F	ਮ	0A2E
ਐ	0A10	ਯ	0A2F
ਓ	0A13	ਰ	0A30
ਔ	0A14	ਲ	0A32
ਕ	0A15	ਵ	0A35
ਖ	0A16	ਸ	0A38
ਗ	0A17	ਹ	0A39
ਘ	0A18	.	0A3C
ਙ	0A19	ਾ	0A3E
ਚ	0A1A	ਿ	0A3F
ਛ	0A1B	ੀ	0A40
ਜ	0A1C	ੁ	0A41
ਝ	0A1D	ੂ	0A42
ਞ	0A1E	ੇ	0A47
ਟ	0A1F	ੈ	0A48
ਠ	0A20	ੋ	0A4B
ਡ	0A21	ਾ	0A4C

Caractere	Código	Caractere	Código
᳚	0A22	᳛	0A4D
᳜	0A23	᳝	0A5C
᳞	0A24	᳟	0A70
᳠	0A25	᳡	0A71
᳢	0A26	᳣	0A72
᳤	0A73		

Conjunto de caracteres em romeno

Para vocabulários personalizados em romeno, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ă	0103	#	0219
â	00E2	#	021B
î	00EE	ș	015F
ț	0163		

Conjunto de caracteres em russo

Para vocabulários personalizados em russo, você pode usar os seguintes caracteres no campo `Phrase`:

Caractere	Código	Caractere	Código
'	0027	п	043F
-	002D	р	0440
.	002E	с	0441
а	0430	т	0442
б	0431	у	0443
в	0432	ф	0444
г	0433	х	0445
д	0434	ц	0446
е	0435	ч	0447
ж	0436	ш	0448
з	0437	щ	0449
и	0438	ъ	044A
й	0439	ы	044B
к	043A	ь	044C
л	043B	э	044D
м	043C	ю	044E
н	043D	я	044F
о	043E	ё	0451

Conjunto de caracteres em sérvio

Para vocabulários personalizados em sérvio, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ć	0107	ı	0456
č	010D	ï	0457
đ	0111	j	0458
š	0161	љ	0459
ž	017E	њ	045A
а	0430	ћ	045B
б	0431	ќ	045C
в	0432	#	045D
г	0433	ђ	045E
д	0434	џ	045F
е	0435	ѓ	0491
ж	0436	ѣ	0493
з	0437	жџ	0497
и	0438	ѝ	0499

Caractere	Código	Caractere	Código
й	0439	ϣ	049B
к	043A	κ	049F
л	043B	κ	04A1
м	043C	ϣ	04A3
н	043D	н	04A5
о	043E	ϣ	04A9
п	043F	ϣ	04AB
р	0440	τ	04AD
с	0441	γ	04AF
т	0442	ϣ	04B1
у	0443	χ	04B3
ф	0444	ϣ	04B5
х	0445	ϣ	04B7
ц	0446	h	04BB
ч	0447	ε	04BD
ш	0448	ϣ	04BF
щ	0449	#	04CA
ъ	044A	ă	04D1
ы	044B	ä	04D3
ь	044C	ě	04D7

Caractere	Código	Caractere	Código
э	044D	ө	04D9
ю	044E	з	04E1
я	044F	й	04E3
#	0450	ö	04E7
ë	0451	ө	04E9
ђ	0452	ÿ	04EF
í	0453	ÿ	04F1
є	0454	ÿ	04F3
s	0455	#	04F7
Ы	04F9		

Conjunto de caracteres em cingalês

Para vocabulários personalizados em cingalês, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
#	0D82	#	0DAF
#	0D83	#	0DB0

Caractere	Código	Caractere	Código
#	0D85	#	0DB1
#	0D86	#	0DB3
#	0D87	#	0DB4
#	0D88	#	0DB5
#	0D89	#	0DB6
#	0D8A	#	0DB7
#	0D8B	#	0DB8
#	0D8C	#	0DB9
#	0D8D	#	0DBA
#	0D91	#	0DBB
#	0D92	#	0DBD
#	0D93	#	0DC0
#	0D94	#	0DC1
#	0D95	#	0DC2
#	0D96	#	0DC3
#	0D9A	#	0DC4
#	0D9B	#	0DC5
#	0D9C	#	0DC6
#	0D9D	#	0DCA
#	0D9E	#	0DCF

Caractere	Código	Caractere	Código
#	0D9F	#	0DD0
#	0DA0	#	0DD1
#	0DA1	#	0DD2
#	0DA2	#	0DD3
#	0DA3	#	0DD4
#	0DA4	#	0DD6
#	0DA5	#	0DD8
#	0DA7	#	0DD9
#	0DA8	##	0DDA
#	0DA9	#	0ddb
#	0DAA	##	0DDC
#	0DAB	###	0DDD
#	0DAC	##	0DDE
#	0DAD	#	0DDF
#	0DAE	#	0DF2

Conjunto de caracteres em eslovaco

Para vocabulários personalizados em eslovaco, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
á	00E1	ň	0148
ä	00E4	ó	00F3
č	010D	ô	00F4
d'	010F	í	0155
é	00E9	š	0161
í	00ED	t'	0165
í	013A	ú	00FA
ĭ	013E	ý	00FD
ž	017E		

Conjunto de caracteres em esloveno

Para vocabulários personalizados em esloveno, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
č	010D	š	0161
ž	017E		

Conjunto de caracteres em somali

Para vocabulários personalizados em somali, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
s	0073	d	0064
t	0074	a	0061
a	0061	r	0072
n	006E	d	0064

Conjunto de caracteres em espanhol

Para vocabulários personalizados em espanhol, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
Á	00C1	á	00E1
É	00C9	é	00E9
Í	00CD	í	00ED
Ó	00D3	ó	0XF3
Ú	00DA	ú	00FA
Ñ	00D1	ñ	0XF1
ü	00FC		

Conjunto de caracteres em sundanês

Para vocabulários personalizados em sundanês, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
s	0073	d	0064
t	0074	a	0061
a	0061	r	0072
n	006E	d	0064

Conjunto de caracteres suaíli

Para vocabulários personalizados em suaíli, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
s	0073	d	0064
t	0074	a	0061
a	0061	r	0072
n	006E	d	0064

Conjunto de caracteres em sueco

Para vocabulários personalizados em sueco, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
Ä	00C4	ä	00E4
Å	00C5	å	00E5
Ö	00D6	ö	00F6

Conjunto de caracteres em tagalo/filipino

Para vocabulários personalizados em tagalo/filipino, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código		
ñ	00F1		

Conjunto de caracteres em tâmil

Para vocabulários personalizados em tâmil, é possível usar os seguintes caracteres no campo `Phrase`:

Caractere	Código	Caractere	Código
அ	0B85	ஈ	0BB0
ஆ	0B86	ஊ	0BB2
இ	0B87	஋	0BB5

Caractere	Código	Caractere	Código
ஈ	0B88	ழ	0BB4
உ	0B89	ள	0BB3
ஊ	0B8A	ற	0BB1
எ	0B8E	ன	0BA9
ஏ	0B8F	ஐ	0B9C
ஐ	0B90	#	0BB6
ஒ	0B92	ஷ	0BB7
ஓ	0B93	ஸ	0BB8
ஔ	0B94	ஹ	0BB9
ஃ	0B83	.	0BCD
க	0B95	ஈ	0BBE
ங	0B99	ற	0BBF
ச	0B9A	ஶ	0BC0
ஞ	0B9E	ழ	0BC1
ட	0B9F	வ	0BC2
ண	0BA3	ஶ	0BC6
த	0BA4	ழ	0BC7
ந	0BA8	ஶ	0BC8
ப	0BAA	ஶ	0BCA
ம	0BAE	ஶ	0BCB

Caractere	Código	Caractere	Código
ш	0BAF	ᄇ	0BCC

Conjunto de caracteres em tártaro

Para vocabulários personalizados em tártaro, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
а	0430	лъ	0459
б	0431	нь	045A
в	0432	ћ	045B
г	0433	ќ	045C
д	0434	#	045D
е	0435	ђ	045E
ж	0436	џ	045F
з	0437	ѓ	0491
и	0438	ѣ	0493
й	0439	жҗ	0497
к	043A	қ	0499

Caractere	Código	Caractere	Código
л	043B	қ	049B
м	043C	к	049F
н	043D	к	04A1
о	043E	ң	04A3
п	043F	н	04A5
р	0440	®	04A9
с	0441	ç	04AB
т	0442	т	04AD
у	0443	ү	04AF
ф	0444	ұ	04B1
х	0445	х	04B3
ц	0446	ц	04B5
ч	0447	ч	04B7
ш	0448	h	04BB
щ	0449	є	04BD
ъ	044A	ё	04BF
ы	044B	#	04CA
ь	044C	ă	04D1
э	044D	ä	04D3
ю	044E	ě	04D7

Caractere	Código	Caractere	Código
я	044F	ə	04D9
#	0450	з	04E1
ë	0451	ñ	04E3
ђ	0452	ö	04E7
í	0453	ө	04E9
є	0454	ÿ	04EF
s	0455	ÿ	04F1
i	0456	ý	04F3
ï	0457	#	04F7
j	0458	Ы	04F9

Conjunto de caracteres em telugo

Para vocabulários personalizados em telugo, é possível usar os seguintes caracteres no campo `Phrase`:

Caractere	Código	Caractere	Código
-	002D	ఁ	0C24
ç	0C01	ఢ	0C25
o	0C02	డ	0C26
ః	0C03	ఢ	0C27
ఁ	0C05	ఞ	0C28
ః	0C06	ఞ	0C2A

Caractere	Código	Caractere	Código
ఇ	0C07	ఫ	0C2B
ఈ	0C08	బ	0C2C
ఉ	0C09	భ	0C2D
ఊ	0C0A	మ	0C2E
ఋ	0C0B	య	0C2F
ఠ	0C30	ఎ	0C0E
ఙ	0C31	ఏ	0C0F
ఠ	0C32	ఐ	0C10
ఠ	0C33	ఒ	0C12
ఠ	0C35	ఓ	0C13
ఠ	0C36	ఔ	0C14
ఠ	0C37	క	0C15
ఠ	0C38	ఖ	0C16
ఠ	0C39	గ	0C17
ఠ	0C3E	ఘ	0C18
ఠ	0C3F	ఙ	0C19
ఠ	0C40	చ	0C1A
ఠ	0C41	ఛ	0C1B
ఠ	0C42	జ	0C1C
ఠ	0C43	ఝ	0C1D

Caractere	Código	Caractere	Código
๔	0C44	๕	0C1E
๖	0C47	๗	0C1F
๘	0C48	๙	0C20
๐	0C4A	๑	0C21
๒	0C4B	๓	0C22
๔	0C4C	๕	0C23
๖	0C4D		

Conjunto de caracteres em tailandês

Para vocabulários personalizados em tailandês, é possível usar os seguintes caracteres no campo `Phrase`:

- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
ก	0E01	ล	0E25
ข	0E02	ฬ	0E26
ฃ	0E03	ว	0E27
ค	0E04	ศ	0E28
ฅ	0E05	ษ	0E29
ฆ	0E06	ส	0E2A

Caractere	Código	Caractere	Código
ง	0E07	ห	0E2B
จ	0E08	ฬ	0E2C
ฉ	0E09	อ	0E2D
ช	0E0A	ฮ	0E2E
ซ	0E0B	๓	0E2F
ฌ	0E0C	๕	0E30
ญ	0E0D	๖	0E31
ฎ	0E0E	๗	0E32
ฏ	0E0F	๘	0E34
ฐ	0E10	๙	0E35
ฑ	0E11	๐	0E36
ฒ	0E12	๑	0E37
ณ	0E13	๒	0E38
ด	0E14	๓	0E39
ต	0E15	๔	0E3A
ถ	0E16	๕	0E40
ท	0E17	๖	0E41
ธ	0E18	๗	0E42
น	0E19	๘	0E43
บ	0E1A	๙	0E44

Caractere	Código	Caractere	Código
ı	0E1B	ġ	0E45
ş	0E1C	ĥ	0E46
ş	0E1D	ı	0E47
ş	0E1E	'	0E48
ş	0E1F	ı	0E49
ğ	0E20	ı	0E4A
ğ	0E21	ı	0E4B
ğ	0E22	ı	0E4C
ğ	0E23	ı	0E4D
ğ	0E24		

Conjunto de caracteres em turco

Para vocabulários personalizados em turco, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
Ç	00C7	ö	00F6

Caractere	Código	Caractere	Código
Ö	00D6	û	00FB
Ü	00DC	ü	00FC
â	00E2	Ǧ	011E
ä	00E4	ǧ	011F
ç	00E7	ı	0130
è	00E8	ı	0131
é	00E9	Ş	015E
ê	00EA	ş	015F
í	00ED	š	0161
î	00EE	ž	017E
ó	00F3		

Conjunto de caracteres em ucraniano

Para vocabulários personalizados em ucraniano, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
a	0430	p	0440

Caractere	Código	Caractere	Código
б	0431	с	0441
в	0432	т	0442
г	0433	у	0443
д	0434	ф	0444
е	0435	х	0445
ж	0436	ц	0446
з	0437	ч	0447
и	0438	ш	0448
й	0439	щ	0449
к	043A	ь	044C
л	043B	ю	044E
м	043C	я	044F
н	043D	ё	0454
о	043E	і	0456
п	043F	ï	0457
ғ	0491		

Conjunto de caracteres em uigur

Para vocabulários personalizados em uigur, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)

- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
#	0611	و	0648
#	0613	س	0649
#	0614	ش	064A
ء	0621	=	064B
آ	0622	ا	064C
أ	0623	=	064D
ؤ	0624	ا	064E
إ	0625	ا	064F
ئ	0626	ا	0650
ا	0627	ا	0651
ب	0628	و	0652
ة	0629	#	0653
ت	062A	#	0654
ث	062B	#	0657
ج	062C	ا	0670
ح	062D	ت	0679
خ	062E	ن	067A
د	062F	ب	067B

Caractere	Código	Caractere	Código
ذ	0630	ت	067C
ر	0631	ت	067D
ز	0632	پ	067E
س	0633	ت	067F
ش	0634	پ	0680
ص	0635	ح	0681
ض	0636	ج	0683
ط	0637	ج	0684
ظ	0638	ح	0685
ع	0639	ج	0686
غ	063A	ج	0687
-	0640	ڈ	0688
ف	0641	د	0689
ق	0642	د	068A
ك	0643	ت	068C
ل	0644	د	068D
م	0645	ت	068F
ن	0646	ژ	0691
ه	0647	ر	0693
ر	0695		

Conjunto de caracteres em uzbeque

Para vocabulários personalizados em uzbeque, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
т	0442	я	044F
б	0431	с	0441
о	043E	ҳ	04B3
п	043F	д	0434
ш	0448	р	0440
и	0438	ў	045E
ч	0447	г	0433
н	043D	ё	0451
қ	049B	й	0439
е	0435	в	0432
ю	044E	э	044D
з	0437	л	043B
х	0445	ф	0444
ц	0446	к	043A

Caractere	Código	Caractere	Código
м	043C	у	0443
ь	044C	ж	0436
ъ	044A	ф	0493
а	0430		

Conjunto de caracteres em vietnamita

O Amazon Transcribe representa os seis tons em vietnamita usando números. A tabela a seguir mostra como as marcas de tom são mapeadas para a palavra “ma”.

Nome do tom	Marca de tom	Número do tom
ngang	ma	ma1
sắc	má	ma2
huyền	mà	ma3
hỏi	mả	ma4
ngã	mã	ma5
nặng	mạ	ma6

Para vocabulários personalizados em vietnamita, você pode usar os seguintes caracteres no campo `Phrase`:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

- & (e comercial)
- ; (ponto e vírgula)
- _ (linha baixa)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
à	00E0	À	00C0
á	00E1	Á	00C1
â	00E2	Â	00C2
ã	00E3	Ã	00C3
è	00E8	È	00C8
é	00E9	É	00C9
ê	00EA	Ê	00CA
ì	00EC	Ì	00CC
í	00ED	Í	00CD
ò	00F2	Ò	00D2
ó	00F3	Ó	00D3
ô	00F4	Ô	00D4
õ	00F5	Õ	00D5
ù	00F9	Ù	00D9
ú	00FA	Ú	00DA
ý	00FD	Ý	00DD

Caractere	Código	Caractere	Código
ǎ	0103	Ǻ	0102
đ	0111	Đ	0110
ĩ	0129	Ĩ	0128
ũ	0169	Ũ	0168
ơ	01A1	Ơ	01A0
ư	01B0	Ư	01AF
ạ	1EA1	Ạ	1EA0
ă	1EA3	Ă	1EA2
ã	1EA5	Ã	1EA4
ą	1EA7	Ą	1EA6
ă	1EA9	Ă	1EA8
ą	1EAB	Ą	1EAA
ậ	1EAD	Ậ	1EAC
ắ	1EAF	Ắ	1EAE
ǎ	1EB1	Ǻ	1EB0
ǎ	1EB3	Ǻ	1EB2
ǎ	1EB5	Ǻ	1EB4
ǎ	1EB7	Ǻ	1EB6
ẹ	1EB9	Ẹ	1EB8
ề	1EBB	Ề	1EBA

Caractere	Código	Caractere	Código
ẽ	1EBD	Ễ	1EBC
ế	1EBF	Ế	1EBE
ề	1EC1	Ề	1EC0
ễ	1EC3	Ễ	1EC2
ễ	1EC5	Ễ	1EC4
ệ	1EC7	Ệ	1EC6
ị	1EC9	Ị	1EC8
ị	1ECB	Ị	1ECA
ọ	1ECD	Ọ	1ECC
ỏ	1ECF	Ỏ	1ECE
ố	1ED1	Ố	1ED0
ồ	1ED3	Ồ	1ED2
ỗ	1ED5	Ỗ	1ED4
ỗ	1ED7	Ỗ	1ED6
ộ	1ED9	Ộ	1ED8
ớ	1EDB	Ớ	1EDA
ờ	1EDD	Ờ	1EDC
ở	1EDF	Ở	1EDE
ỡ	1EE1	Ỡ	1EE0
ợ	1EE3	Ợ	1EE2

Caractere	Código	Caractere	Código
ұ	1EE5	Ў	1EE4
ů	1EE7	Ў̇	1EE6
ú	1EE9	Ў̇̇	1EE8
ù	1EEB	Ў̇̇̇	1EEA
ǔ	1EED	Ў̇̇̇̇	1EEC
ũ	1EEF	Ў̇̇̇̇̇	1EEE
ұ	1EF1	Ў̇̇̇̇̇̇	1EF0
ỳ	1EF3	Ỳ	1EF2
Ʒ	1EF5	Ʒ	1EF4
ỷ	1EF7	Ỳ̇	1EF6
ỹ	1EF9	Ỳ̇̇	1EF8

Conjunto de caracteres em galês

Para vocabulários personalizados em galês, é possível usar os seguintes caracteres no campo Phrase:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo Phrase:

Caractere	Código	Caractere	Código
à	00E0	ò	00F2

Caractere	Código	Caractere	Código
á	00E1	ó	00F3
â	00E2	ô	00F4
ä	00E4	ö	00F6
è	00E8	ù	00F9
é	00E9	ú	00FA
ê	00EA	û	00FB
ë	00EB	ü	00FC
ì	00EC	ý	00FD
í	00ED	ÿ	00FF
î	00EE	ŵ	0175
ï	00EF	ÿ	0177
ÿ	1EF3		

Conjunto de caracteres em uolofe

Para vocabulários personalizados em uolofe, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
à	00E0	ê	00EA
ã	00E3	ë	00EB
ç	00E7	ñ	00F1
è	00E8	ó	00F3
é	00E9	ô	00F4
η	014B		

Conjunto de caracteres em zulu

Para vocabulários personalizados em zulu, é possível usar os seguintes caracteres no campo `Phrase`:

- a - z
- - (hífen)
- . (ponto final)

Também é possível usar os seguintes caracteres Unicode no campo `Phrase`:

Caractere	Código	Caractere	Código
s	0073	d	0064
t	0074	a	0061
a	0061	r	0072
n	006E	d	0064

Como o Amazon Transcribe funciona

Amazon Transcribe usa modelos de aprendizado de máquina para converter fala em texto.

Além do texto transcrito, as transcrições contêm dados sobre o conteúdo transcrito, incluindo pontuações de confiança e registros de data e hora para cada palavra ou sinal de pontuação. Para ver um exemplo de saída, consulte a seção [Entrada e saída de dados](#). Para obter uma lista completa dos recursos que você pode aplicar à sua transcrição, consulte o [resumo do recurso](#).

Os métodos de transcrição podem ser separados em duas categorias principais:

- **Transcrições Batch:** Transcribe arquivos de mídia que foram enviados para um Amazon S3 bucket. Você pode usar o [AWS CLI/AWS Management Console](#), e vários [AWS SDKs](#) para transcrições em lote.
- **Transcrições de streaming:** Transcribe fluxos de mídia em tempo real. Você pode usar o [AWS Management Console](#), [HTTP/2](#) e vários [AWS SDKs](#) para transmitir transcrições. [WebSockets](#)

Observe que o suporte a recursos e idiomas difere para transcrições em lote e streaming. Para obter mais informações, consulte [Amazon Transcribe features](#) e [Idiomas suportados](#).

Tópicos

- [Entrada e saída de dados](#)
- [Transcrevendo números e pontuação](#)

Operações de API para você começar

Batch: [StartTranscriptionJob](#)

Transmissão: [StartStreamTranscription](#), [StartStreamTranscriptionWebSocket](#)

Entrada e saída de dados

Amazon Transcribe pega dados de áudio, como um arquivo de mídia em um Amazon S3 bucket ou fluxo de mídia, e os converte em dados de texto.

Se você estiver transcrevendo arquivos de mídia armazenados em um Amazon S3 bucket, estará realizando transcrições em lote. Se você estiver transcrevendo streams de mídia, você está realizando transcrições de streaming. Esses dois processos têm regras e requisitos diferentes.

Com as transcrições em lote, você pode usar [Fila de Job](#) se não precisar processar todos os seus trabalhos de transcrição simultaneamente. Isso Amazon Transcribe permite acompanhar seus trabalhos de transcrição e processá-los quando houver vagas disponíveis.

Note

Amazon Transcribe pode armazenar temporariamente seu conteúdo para melhorar continuamente a qualidade de seus modelos de análise. Consulte as [Perguntas frequentes do Amazon Transcribe](#) para saber mais. Para solicitar a exclusão de conteúdo que possa ter sido armazenado por Amazon Transcribe, abra um caso com [AWS Support](#).

Tópicos

- [Formatos de mídia](#)
- [Canais de áudio](#)
- [Taxas de amostragem](#)
- [Resultado](#)

Formatos de mídia

Os tipos de mídia compatíveis diferem entre transcrições em lote e transcrições de streaming, embora formatos sem perdas sejam recomendados para ambas. Para detalhes, consulte a tabela a seguir:

	Lote	Streaming
Formatos com suporte	<ul style="list-style-type: none"> • BRAÇO • FLAC • M4A • MP3 • MP4 	<ul style="list-style-type: none"> • FLAC • Ogg Opus • Codificação PCM

	Lote	Streaming
	<ul style="list-style-type: none"> • Ogg • WebM • WAV 	
Formatos recomendados	<ul style="list-style-type: none"> • FLAC • WAV com codificação PCM de 16 bits 	<ul style="list-style-type: none"> • FLAC • Áudio little-endian de 16 bits assinado pelo PCM (observe que isso não inclui WAV)

Para obter melhores resultados, use um formato sem perdas, como FLAC ou WAV com codificação PCM de 16 bits.

Note

As transcrições de streaming não são suportadas em todos os idiomas. Consulte a coluna “Entrada de dados” na [tabela de idiomas suportados](#) para obter detalhes.

Canais de áudio

Amazon Transcribe suporta mídia de canal único e canal duplo. Atualmente, não há suporte para mídia com mais de dois canais.

Se o áudio contiver vários alto-falantes em um canal e você quiser particionar e rotular cada alto-falante em sua saída de transcrição, você pode usar o [particionamento de alto-falante \(diarização\)](#).

Se o áudio contiver fala em dois canais separados, você pode usar a [identificação do canal](#) para transcrever cada canal separadamente em sua transcrição.

Ambas as opções produzem um arquivo de transcrição.

Note

Se você não habilitar o [particionamento de alto-falantes](#) ou a [identificação de canais](#), o texto da transcrição será fornecido como uma seção contínua.

Taxas de amostragem

Com trabalhos de transcrição em lote, você pode optar por fornecer uma taxa de amostragem, embora esse parâmetro seja opcional. Se você incluí-lo em sua solicitação, certifique-se de que o valor fornecido corresponda à taxa de amostragem real em seu áudio. Se você fornecer uma taxa de amostragem que não corresponda ao seu áudio, seu trabalho poderá falhar.

Com transcrições de streaming, você deve incluir uma taxa de amostragem em sua solicitação. Assim como nas tarefas de transcrição em lote, certifique-se de que o valor fornecido corresponda à taxa de amostragem real em seu áudio.

As taxas de amostragem para áudio de baixa fidelidade, como gravações telefônicas, normalmente usam 8.000 Hz. Para áudio de alta fidelidade, Amazon Transcribe suporta valores entre 16.000 Hz e 48.000 Hz.

Resultado

A saída da transcrição é no formato JSON. A primeira parte da sua transcrição contém a própria transcrição em forma de parágrafo, seguida por dados adicionais para cada palavra e sinal de pontuação. Os dados fornecidos dependem dos recursos que você inclui em sua solicitação. No mínimo, sua transcrição contém a hora de início, a hora de término e a pontuação de confiança de cada palavra. A [seção a seguir](#) mostra um exemplo de saída de uma solicitação básica de transcrição que não incluiu nenhuma opção ou recurso adicional.

Todas as transcrições em lote são armazenadas em Amazon S3 baldes. Você pode optar por salvar sua transcrição em seu próprio Amazon S3 bucket ou Amazon Transcribe usar um bucket padrão seguro. Para saber mais sobre como criar e usar Amazon S3 buckets, consulte [Como trabalhar com buckets](#).

Se você quiser que sua transcrição seja armazenada em um Amazon S3 bucket de sua propriedade, especifique o URI do bucket em sua solicitação de transcrição. Certifique-se de conceder permissões de Amazon Transcribe gravação para esse bucket antes de iniciar seu trabalho de transcrição em lote. Se você especificar seu próprio bucket, sua transcrição permanecerá nesse bucket até que você a remova.

Se você não especificar um Amazon S3 bucket, Amazon Transcribe usa um bucket seguro gerenciado por serviços e fornece um URI temporário que você pode usar para baixar sua transcrição. Observe que os URIs temporários são válidos por 15 minutos. Se você receber um `AccessDenied` erro ao usar o URI fornecido, faça uma `GetTranscriptionJob` solicitação para obter um novo URI temporário para sua transcrição.

Se você optar por um bucket padrão, sua transcrição será excluída quando seu trabalho expirar (90 dias). Se você quiser manter sua transcrição após essa data de validade, você deve baixá-la.

As transcrições de streaming são retornadas pelo mesmo método que você está usando para sua transmissão.

Tip

Se você quiser converter sua saída JSON em uma turn-by-turn transcrição no formato Word, veja este [GitHub exemplo \(para Python3\)](#). Esse script funciona com transcrições de análise pós-chamada e transcrições de lote padrão com a diarização ativada.

Exemplo de saída

As transcrições fornecem uma transcrição completa em forma de parágrafo, seguida por um word-for-word detalhamento, que fornece dados para cada palavra e sinal de pontuação. Isso inclui a hora de início, a hora de término, uma pontuação de confiança e um tipo (`pronunciation` ou `punctuation`).

O exemplo a seguir é de um trabalho simples de transcrição em lote que não incluiu nenhum [recurso adicional](#). Com cada recurso adicional que você aplica à sua solicitação de transcrição, você obtém dados adicionais no arquivo de saída da transcrição.

As transcrições básicas em lote contêm duas seções principais:

1. `transcripts`: contém toda a transcrição em um bloco de texto.
2. `items`: contém informações sobre cada palavra e sinal de pontuação da `transcripts` seção.

Cada recurso adicional que você inclui em sua solicitação de transcrição produz informações adicionais em sua transcrição.

```
{
  "jobName": "my-first-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "Welcome to Amazon Transcribe."
      }
    ]
  }
}
```

```
],
  "items": [
    {
      "start_time": "0.64",
      "end_time": "1.09",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "Welcome"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "1.09",
      "end_time": "1.21",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "to"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "1.21",
      "end_time": "1.74",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "Amazon"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "1.74",
      "end_time": "2.56",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "Transcribe"
        }
      ]
    }
  ],
```



```

        "type": "pronunciation"
    },
    {
        "alternatives": [
            {
                "confidence": "0.0",
                "content": "."
            }
        ],
        "type": "punctuation"
    }
]
},
"status": "COMPLETED"
}

```

Transcrevendo números e pontuação

Amazon Transcribe adiciona pontuação automaticamente a todos os idiomas suportados e coloca palavras em maiúsculas de forma adequada para idiomas que usam distinção de maiúsculas e minúsculas em seus sistemas de escrita.

Para a maioria dos idiomas, os números são transcritos em suas formas de palavras. No entanto, se sua mídia estiver em inglês ou alemão, Amazon Transcribe trate os números de forma diferente, dependendo do contexto em que são usados.

Por exemplo, se um orador disser “Meet me at eight-thirty AM on June first at one-hundred Main Street with three-dollars-and-fifty-cents and one-point-five chocolate bars, isso é transcrito como:

- Dialectos inglês e alemão: Meet me at 8:30 a.m. on June 1st at 100 Main Street with \$3.50 and 1.5 chocolate bars
- Todos os outros idiomas: Meet me at eight thirty a m on June first at one hundred Main Street with three dollars and fifty cents and one point five chocolate bars

Para ver todas as regras associadas aos números falados em inglês e alemão, consulte a tabela a seguir.

Regras	Dialeto inglês (entrada de áudio → texto de saída)	Dialeto alemão (entrada de áudio → texto de saída)
Converta números cardinais maiores que dez em números.	<ul style="list-style-type: none"> • "Fifty five" → 55 • "a hundred" → 100 • "One thousand and thirty one" → 1031 • "One hundred twenty-three million four hundred fifty six thousand seven hundred eight nine" → 123,456,789 	<ul style="list-style-type: none"> • "fünfundfünfzig" → 55 • "vier tausend sechs hundert einundachtzig" → 4681 • "eine Sache" → "eine Sache"
Converter números cardinais seguidos de “milhão” ou “bilhão” em numerais seguidos de uma palavra quando “milhão” ou “bilhão” não for seguido por um número.	<ul style="list-style-type: none"> • "one hundred million" → 100 million • "one billion" → 1 billion • "two point three million" → 2.3 million 	<ul style="list-style-type: none"> • "zehn Millionen Menschen" → 10 Millionen Menschen • "zehn Millionen fünf hundert tausend" → 10.500.000
Converta números ordinais maiores que dez em números.	<ul style="list-style-type: none"> • "Forty third" → 43rd • "twenty sixth avenue" → 26th avenue 	<ul style="list-style-type: none"> • "dreiundzwanzigste" → 23 • "vierzigster" → 40 • "ich war Erster" → "ich war Erster"
Converter frações no respectivo formato numérico.	<ul style="list-style-type: none"> • "a quarter" → 1/4 • "three sixteenths" → 3/16 • "a half" → 1/2 • "a hundredth" → 1/100 	As frações não são convertidas em um formato numérico. <ul style="list-style-type: none"> • "ein Drittel" → "ein Drittel"
Converta números menores que dez em dígitos se houver mais de um em uma linha.	<ul style="list-style-type: none"> • "three four five" → 345 • "My phone number is four two five five five five one two one two" → My phone number is 4255551212 	<ul style="list-style-type: none"> • "eins zwei drei" → 123 • "plus vier neun zwei vier eins" → +49241

Regras	Dialectos ingleses (entrada de áudio → texto de saída)	Dialectos alemães (entrada de áudio → texto de saída)
As palavras “ponto” ou “ponto” são exibidas como decimal.	<ul style="list-style-type: none"> • "three hundred and three dot five" → 303.5 • "three point twenty three" → 3.23 • "zero point four" → 0.4 • "point three" → 0.3 	Os decimais são indicados por “, ”. <ul style="list-style-type: none"> • "zweiundzwanzig komma drei" → 22,3
Converter a palavra “por cento” após um número em um símbolo de porcentagem (%).	<ul style="list-style-type: none"> • "twenty three percent" → 23% • "twenty three point four five percent" → 23.45% 	<ul style="list-style-type: none"> • "fünf Prozent Hürde" → 5% Hürde • "dreiundzwanzig komma vier Prozent" → 23,4%

Regras	Dialeto inglês (entrada de áudio → texto de saída)	Dialeto alemão (entrada de áudio → texto de saída)
<p>Converta palavras monetárias em símbolos.</p>	<p>Converta as palavras “dólar”, “dólar americano”, “dólar australiano”, “AUD” ou “USD” depois de um número em um cifrão (\$) antes do número.</p> <ul style="list-style-type: none"> • "one dollar and fifteen cents" → \$1.15 • "twenty three USD" → \$23 • "twenty three Australian dollars" → \$23 <p>Converter as palavras “libras”, “libras esterlinas” ou “GDB” após um número em sinal de libra (£) antes do número.</p> <ul style="list-style-type: none"> • "twenty three pounds" → £23 • "I have two thousand pounds" → I have £2,000 • "five pounds thirty three pence" → £5.33 <p>Converter as palavras “rúpias”, “rúpias indianas” ou “INR” após um número no símbolo de rúpia (#) antes do número.</p> <ul style="list-style-type: none"> • "twenty three rupees" → #23 	<p>Converta as palavras “Euro” em um sinal de euro.</p> <ul style="list-style-type: none"> • "ein euro" → 1 € • "ein Euro vierzig" → 1,40 € • "ein Euro vierzig Cent" → 1,40 €

Regras	Dialeto inglês (entrada de áudio → texto de saída)	Dialeto alemão (entrada de áudio → texto de saída)
	<ul style="list-style-type: none"> "fifty rupees thirty paise" → #50.30 	
Converter horas em números.	<ul style="list-style-type: none"> "seven a m eastern standard time" → 7 a.m. eastern standard time "twelve thirty p m" → 12:30 p.m. 	<ul style="list-style-type: none"> "vierzehn Uhr fünfzehn" → 14:15 Uhr
Converter datas em números.	<ul style="list-style-type: none"> "May fifth twenty twelve" → May 5th 2012 "May five twenty twelve" → May 5 2012 "five May twenty twelve" → 5 May 2012 	<ul style="list-style-type: none"> "dritter Dezember neunzehn hundert sechundfünfzig" → 3. Dezember 1956
Separe os intervalos de números pela palavra "para".	<ul style="list-style-type: none"> "twenty three to thirty seven" → 23 to 37 	Não aplicável
Os anos são representados como quatro dígitos; isso só é válido para anos nos séculos XX, XXI e 22°.	<ul style="list-style-type: none"> "nineteen sixty two" → 1962 "the year is twenty twelve" → o ano é 2012 "twenty nineteen" → 2019 "twenty one thirty" → 2130 	Não aplicável
Exibir barras e traços.	<ul style="list-style-type: none"> "fifty-five dash thirteen" → 55-13 <p>As barras não são exibidas.</p> <ul style="list-style-type: none"> "fifty-five slash thirteen" → 55 slash 13 	<ul style="list-style-type: none"> "fünfundfünfzig Schrägstrich dreizehn" → 55/13 "fünfundfünfzig Strich dreizehn" → 55-13

Regras	Dialeto inglês (entrada de áudio → texto de saída)	Dialeto alemão (entrada de áudio → texto de saída)
Exibir parágrafos numerados.	Os parágrafos numerados não são exibidos usando o símbolo do parágrafo (§). <ul style="list-style-type: none">• "paragraph seventeen" → paragraph 17	<ul style="list-style-type: none">• "Paragraf siebzehn" → § 17

Conceitos básicos do Amazon Transcribe

Antes de criar transcrições, você tem alguns pré-requisitos:

- [Cadastrar-se com uma Conta da AWS](#)
- [Instale os SDKs AWS CLI e](#) (se você estiver usando o AWS Management Console para suas transcrições, você pode pular esta etapa)
- [Configurar IAM credenciais](#)
- [Configurar um Amazon S3 bucket](#)
- [Crie uma IAM política](#)

Depois de concluir esses pré-requisitos, você estará pronto para transcrever. Selecione seu método de transcrição preferido na lista a seguir para começar.

- [AWS CLI](#)
- [AWS Management Console](#)
- [AWS SDK](#)
- [HTTP](#)
- [WebSockets](#)

Tip

Se você é novo no Amazon Transcribe ou gostaria de explorar nossos recursos, recomendamos usar [AWS Management Console](#). Essa também é a opção mais fácil se você quiser iniciar uma transmissão usando o microfone do seu computador.

Como o streaming usando HTTP/2 WebSockets é mais complicado do que os outros métodos de transcrição, recomendamos revisar a [Configurar uma transcrição de streaming](#) seção antes de começar a usar esses métodos. Observe que é altamente recomendável usar um SDK para transcrições de streaming.

Cadastrar-se em uma Conta da AWS

Você pode se inscrever para uma conta de [nível gratuito](#) ou uma [conta paga](#). Ambas as opções oferecem acesso a todos os Serviços da AWS. O nível gratuito tem um período de teste durante o qual você pode explorar os Serviços da AWS e estimar seu uso. Quando o período de teste expirar, você poderá migrar para uma conta paga. As taxas são acumuladas com base em uma pay-as-you-use base; consulte [Amazon Transcribe Preços](#) para obter detalhes.

Tip

Ao configurar sua conta, anote seu ID da Conta da AWS porque você precisa dele para criar entidades IAM.

Instalando os SDKs AWS CLI e

Para usar a Amazon Transcribe API, você deve primeiro instalar o AWS CLI. A atual AWS CLI é a versão 2. Você pode encontrar instruções de instalação para [Linux](#), [Mac](#), [Windows](#) e [Docker](#) no [Guia do AWS Command Line Interface usuário](#).

Depois de instalar o AWS CLI, você deve [configurá-lo](#) para suas credenciais de segurança da Região da AWS.

Se você quiser usar o Amazon Transcribe com um SDK, selecione seu idioma preferido para obter instruções de instalação:

- [.NET](#)
- [C++](#)
- [Go](#)
- [Java 2](#)
- [JavaScript](#)
- [PHP 3](#)
- [AWS SDK for Python \(Boto3\)](#) (transcrições em lote)
- [Python](#) (transcrições de streaming)
- [Ruby 3](#)
- [Ferrugem](#) (transcrições em lote)

- [Rust](#) (transcrições em streaming)

Configurar IAM credenciais

Ao criar uma Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos os AWS serviços e recursos em sua conta. Essa identidade é chamada de usuário Conta da AWS raiz e é acessada pelo login com o endereço de e-mail e a senha que você usou para criar a conta.

É altamente recomendável não usar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente ele pode executar.

Como prática recomendada, exija que os usuários, inclusive os que precisam de acesso de administrador, usem a federação com um provedor de identidades AWS para acessar a usando credenciais temporárias.

Uma identidade federada é qualquer usuário que acessa os AWS serviços usando credenciais fornecidas por uma fonte de identidades. Quando as identidades federadas acessam Contas da AWS, elas assumem funções que fornecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos usar o [AWS IAM Identity Center](#). Você pode criar usuários e computadores em IAM Identity Center. Ou você pode se conectar e sincronizar com um conjunto de usuários e grupos em sua própria fonte de identidade para uso em todas as suas Contas da AWS e aplicações. Para obter mais informações, consulte [Identity and Access Management para Amazon Transcribe](#).

Para saber mais sobre as IAM melhores práticas, consulte [Práticas recomendadas de segurança em IAM](#).

Criação de um Amazon S3 bucket

Amazon S3 é um serviço seguro de armazenamento de objetos. Amazon S3 armazena seus arquivos (chamados objetos) em contêineres (chamados de buckets).

Para executar uma transcrição em lote, primeiro você deve carregar seus arquivos de mídia em um Amazon S3 bucket. Se você não especificar um Amazon S3 bucket para sua saída de transcrição, Amazon Transcribe colocará sua transcrição em um Amazon S3 bucket AWS gerenciado temporariamente. A saída da transcrição AWS em buckets gerenciados é excluída automaticamente após 90 dias.

Saiba como [criar seu primeiro bucket do S3](#) e [fazer upload de um objeto para seu bucket](#).

Criar uma política do IAM

Para gerenciar o AWS, você deve criar políticas e anexando-as a IAM identidades do (usuários, usuários, usuários, grupos ou funções) ou AWS recursos. Uma política define as permissões da entidade à qual está vinculada. Por exemplo, uma função só pode acessar um arquivo de mídia localizado em seu Amazon S3 bucket se você tiver anexado uma política a essa função que conceda acesso a ela. Se quiser restringir ainda mais essa função, você pode limitar seu acesso a um arquivo específico dentro Amazon S3 de um bucket.

Para saber mais sobre como usar AWS políticas

- [Políticas e permissões em IAM](#)
- [Criação de IAM políticas](#)
- [Como Amazon Transcribe funciona com o IAM](#)

Por exemplo, políticas que você pode usar com Amazon Transcribe, consulte [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#). Se você quiser gerar políticas personalizadas, considere usar o [Gerador AWS de políticas](#).

Você pode adicionar uma política usando o AWS Management Console AWS CLI, ou AWS SDK. Para obter instruções, consulte Como [adicionar e remover permissões de IAM identidade](#).

As políticas têm o formato:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "my-policy-name",
      "Effect": "Allow",
      "Action": [
        "service:action"
      ],
      "Resource": [
        "amazon-resource-name"
      ]
    }
  ]
}
```

```
}

```

Os nomes de recursos da Amazon (ARNs) identificam de forma exclusiva todos os AWS recursos, como um Amazon S3 bucket. Você pode usar ARNs em sua política para conceder permissões para ações específicas para usar recursos específicos. Por exemplo, se você quiser conceder acesso de leitura a um Amazon S3 bucket e suas subpastas, você pode adicionar o seguinte código à `Statement` seção da sua política de confiança:

```
{
  "Effect": "Allow",
  "Action": [
    "s3:GetObject",
    "s3:ListBucket"
  ],
  "Resource": [
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
  ]
}
```

Aqui está um exemplo de política que concede permissões de Amazon Transcribe leitura (`GetObject`, `ListBucket`, `PutObject`) e gravação (`PutObject`) a um Amazon S3 bucket e suas subpastas: `DOC-EXAMPLE-BUCKET`

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

```

    ],
    "Resource": [
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    ]
  }
]
}

```

Transcrevendo com oAWS Management Console

Você pode usar oAWS console para transcrições em lote e streaming. Se você estiver transcrevendo um arquivo de mídia localizado em umAmazon S3 bucket, você está realizando uma transcrição em lote. Se você estiver transcrevendo um fluxo de dados de áudio em tempo real, você está realizando uma transcrição de streaming.

Antes de iniciar uma transcrição em lote, você deve primeiro carregar seu arquivo de mídia em umAmazon S3 bucket. Para transmitir transcrições usando oAWS Management Console, você deve usar o microfone do seu computador.

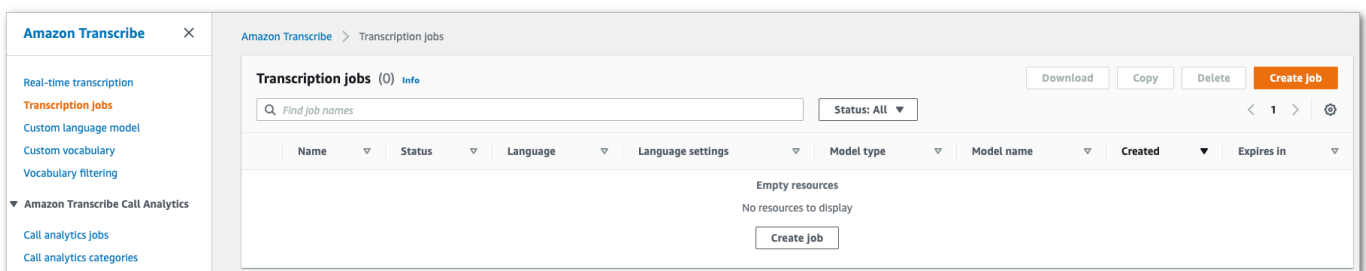
Para ver os formatos de mídia compatíveis e outros requisitos e restrições de mídia, consulte [Entrada e saída de dados](#).

Expanda as seções a seguir para obter breves orientações sobre cada método de transcrição.

Transcrições em Batch

Primeiro, certifique-se de ter carregado o arquivo de mídia que deseja transcrever em umAmazon S3 bucket. Se você não tiver certeza de como fazer isso, consulte o Guia doAmazon S3 usuário: [Carregue um objeto em seu bucket](#).

1. Em [AWS Management Console](#), selecione Trabalhos de transcrição no painel de navegação esquerdo. Isso leva você a uma lista de seus trabalhos de transcrição.



Selecione Criar trabalho.

2. Preencha os campos na página Especificar detalhes do trabalho.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)

If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)

If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

▶ **Additional settings**

O local de entrada deve ser um objeto dentro de um Amazon S3 bucket. Para o local de saída, você pode escolher um bucket seguro Amazon S3 gerenciado por serviços ou especificar seu próprio Amazon S3 bucket.

Se você escolher um bucket gerenciado por serviços, poderá ver uma prévia da transcrição no AWS Management Console e fazer o download da transcrição na página de detalhes do trabalho (veja abaixo).

Se você escolher seu próprio Amazon S3 bucket, não poderá ver uma prévia no AWS Management Console e deverá acessar o Amazon S3 bucket para baixar sua transcrição.

Input data [Info](#)

Input file location on S3
Choose an input audio or video file in Amazon S3.

Valid file formats: MP3, MP4, WAV, FLAC, AMR, OGG, and WebM.

Output data

Output data location type info [Info](#)

Service-managed S3 bucket
The output will be removed after 90 days when the job expires.

Customer specified S3 bucket
The output will not be removed from bucket even after the job expires.

Subtitle file format [Info](#)

SRT (SubRip)

VTT (WebVTT)

Tags - optional

A tag is a label you can add to a resource as metadata to help you organize, search, or filter your data. Each tag consists of a key and an optional value, in the form 'key:value'.

No tags associated with the resource.

You can add up to 50 more tags.

Selecione Next (Próximo).

3. Selecione as opções desejadas na página Configurar tarefa. Se você quiser usar [Vocabulários personalizados](#) ou [Modelos de linguagem personalizados](#) com sua transcrição, você deve criá-los antes de iniciar seu trabalho de transcrição.

Configure job - *optional* [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

Automatic content redaction [Info](#)
Automatic content redaction removes personally identifiable information (PII) in your transcripts. Redactions in transcripts show up as [PII].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

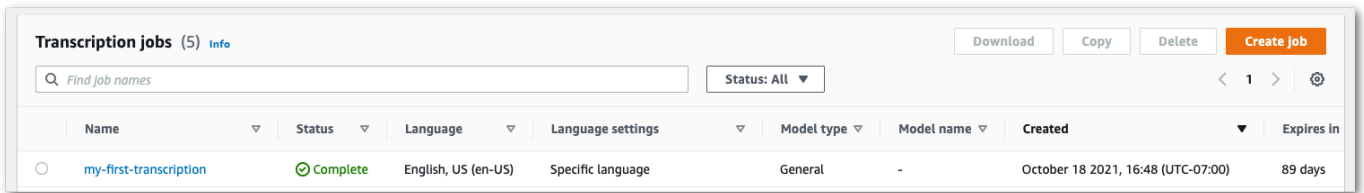
Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

[Cancel](#) [Previous](#) [Create job](#)

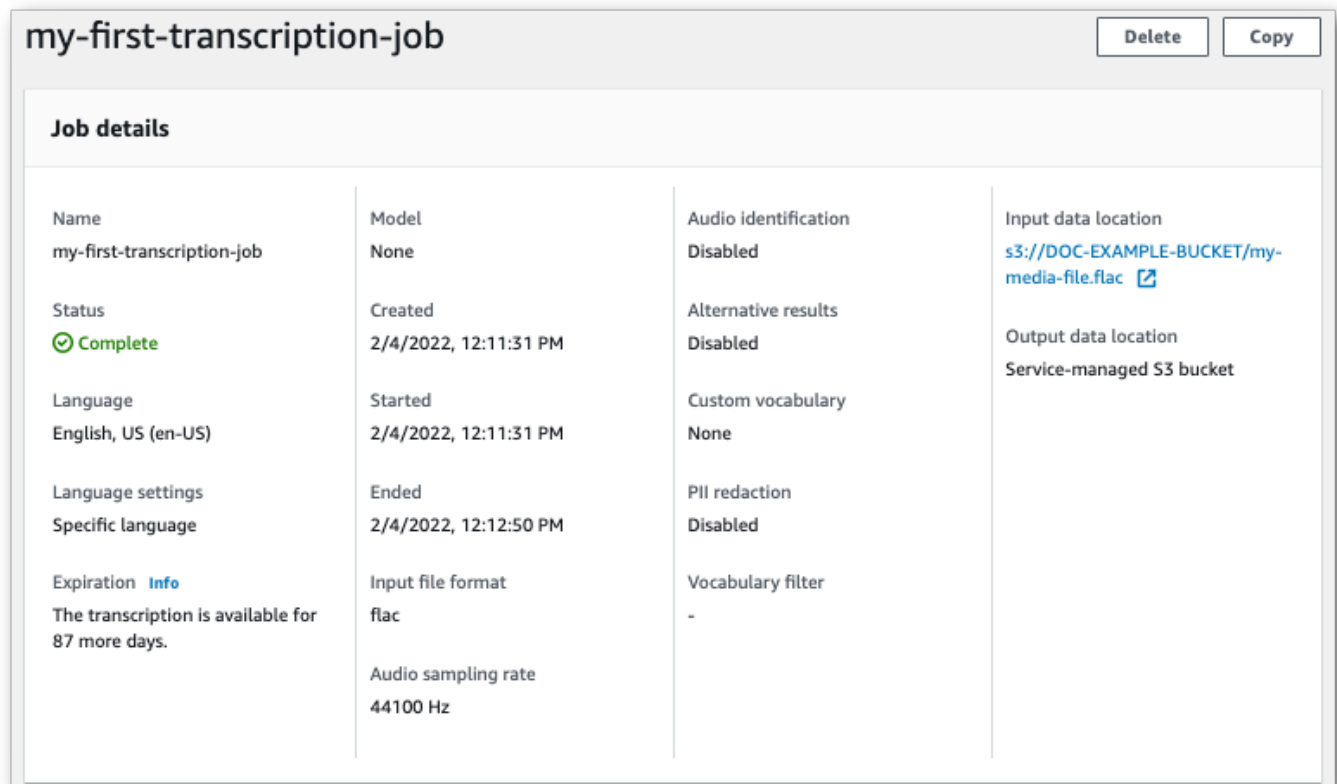
Selecione Criar trabalho.

4. Agora você está na página de trabalhos de transcrição. Aqui você pode ver o status do trabalho de transcrição. Depois de concluir, selecione sua transcrição.



5. Agora você está visualizando a página de detalhes do Job para sua transcrição. Aqui você pode ver todas as opções especificadas ao configurar seu trabalho de transcrição.

Para ver sua transcrição, selecione o caminho do arquivo vinculado na coluna à direita em Local dos dados de saída. Isso leva você para a pasta Amazon S3 de saída que você especificou. Selecione seu arquivo de saída, que agora tem uma extensão.json.



6. A forma como você baixa sua transcrição depende de você ter escolhido um Amazon S3 bucket gerenciado por serviços ou seu próprio Amazon S3 bucket.
- Se você escolher um bucket gerenciado por serviços, poderá ver um painel de visualização da transcrição na página de informações da tarefa de transcrição, junto com o botão Download.

my-first-transcription-job

Job details

Name my-first-transcription-job	Model None	Audio identification Disabled	Input data location s3://DOC-EXAMPLE-BUCKET/my-media-file.flac
Status ✔ Complete	Created 2/4/2022, 12:11:31 PM	Alternative results Disabled	Output data location Service-managed S3 bucket
Language English, US (en-US)	Started 2/4/2022, 12:11:31 PM	Custom vocabulary None	
Language settings Specific language	Ended 2/4/2022, 12:12:50 PM	PII redaction Disabled	
Expiration Info The transcription is available for 87 more days.	Input file format flac	Vocabulary filter -	
	Audio sampling rate 44100 Hz		

Transcription preview ▼

You can see the first 5,000 characters of the transcription text below. To download the full text, choose Download full transcript.

Text
Audio identification
Subtitles

This is a preview of the content of your transcript. If your transcript is long, you may have to scroll to see the complete preview.

Selecione Baixar e escolha Baixar transcrição.

- b. Se você escolheu seu próprio Amazon S3 bucket, não verá nenhum texto no painel de visualização da transcrição na página de informações do seu trabalho de transcrição. Em vez disso, você vê uma caixa de informações azul com um link para o Amazon S3 bucket que você escolheu.

my-first-transcription-job

Job details

Name my-first-transcription-job	Model None	Audio identification Disabled	Input data location s3://DOC-EXAMPLE-BUCKET/my-media-file.flac ↗
Status ✔ Complete	Created 2/7/2022, 11:42:17 AM	Alternative results Disabled	Output data location https://s3.us-west-2.amazonaws.com/DOC-EXAMPLE-BUCKET ↗
Language English, US (en-US)	Started 2/7/2022, 11:42:17 AM	Custom vocabulary None	
Language settings Specific language	Ended 2/7/2022, 11:43:37 AM	PII redaction Disabled	
Expiration Info The transcription is available for 89 more days.	Input file format flac	Vocabulary filter -	
	Audio sampling rate 44100 Hz		

Transcription preview ▼

Select download to save a local copy of the transcription.

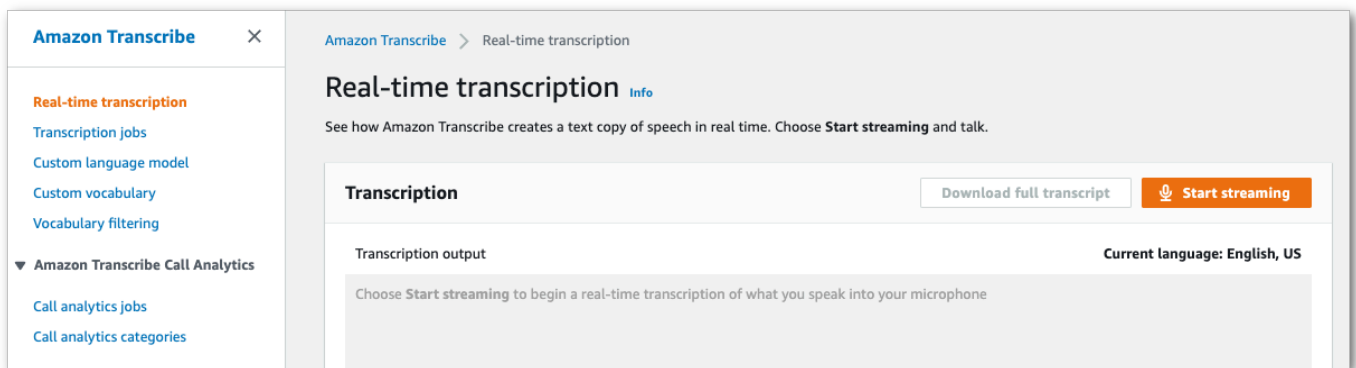
Text
Audio identification
Subtitles

ⓘ When you use your own S3 bucket for transcription output, Amazon Transcribe does not show the output in the console. You open the output file from your [S3 Bucket](#). [↗](#)

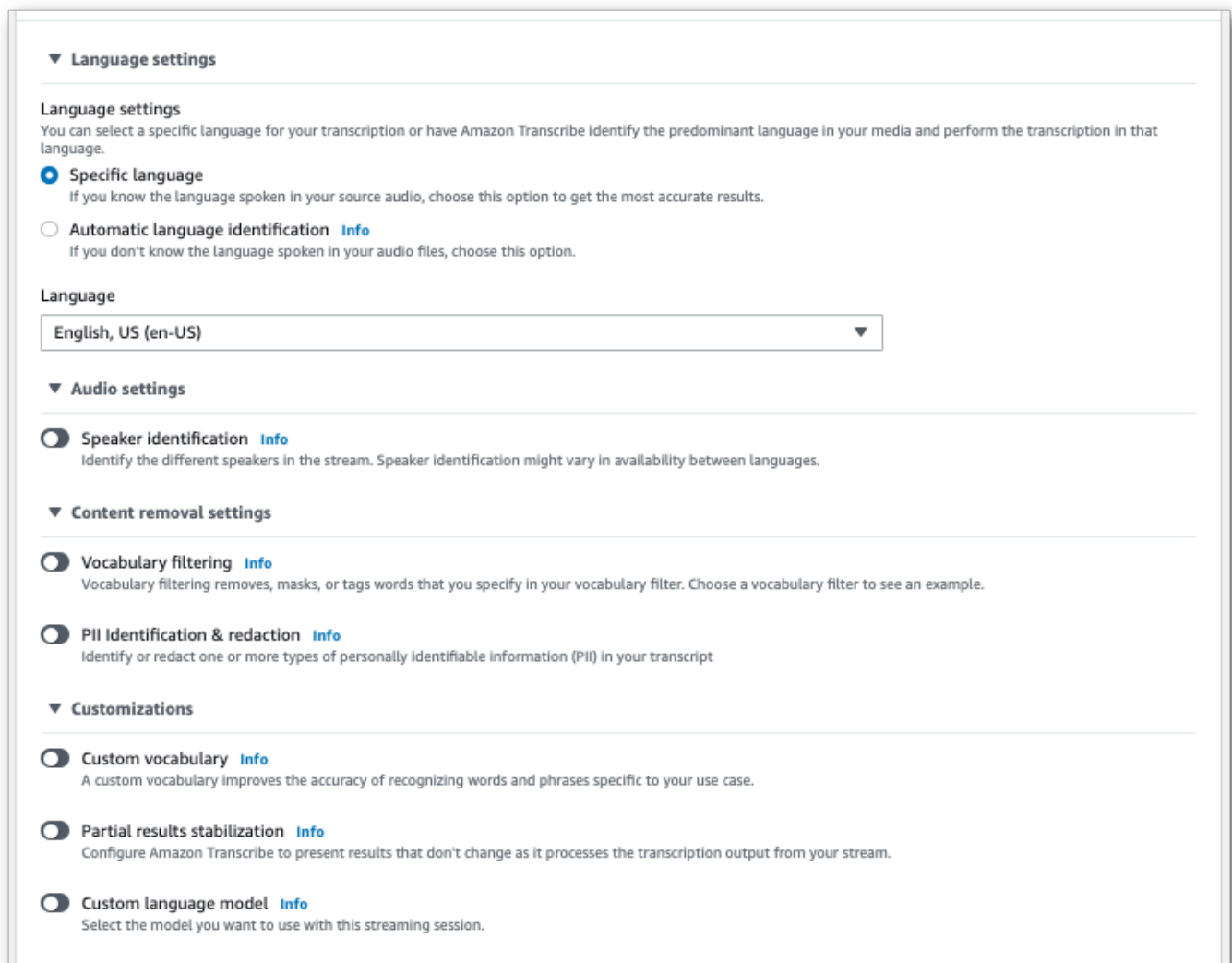
Para acessar sua transcrição, acesse o Amazon S3 bucket especificado usando o link em Local de saída dos dados no painel Detalhes do Job ou o link do S3 Bucket na caixa de informações azul no painel de visualização da transcrição.

Transcrições de streaming

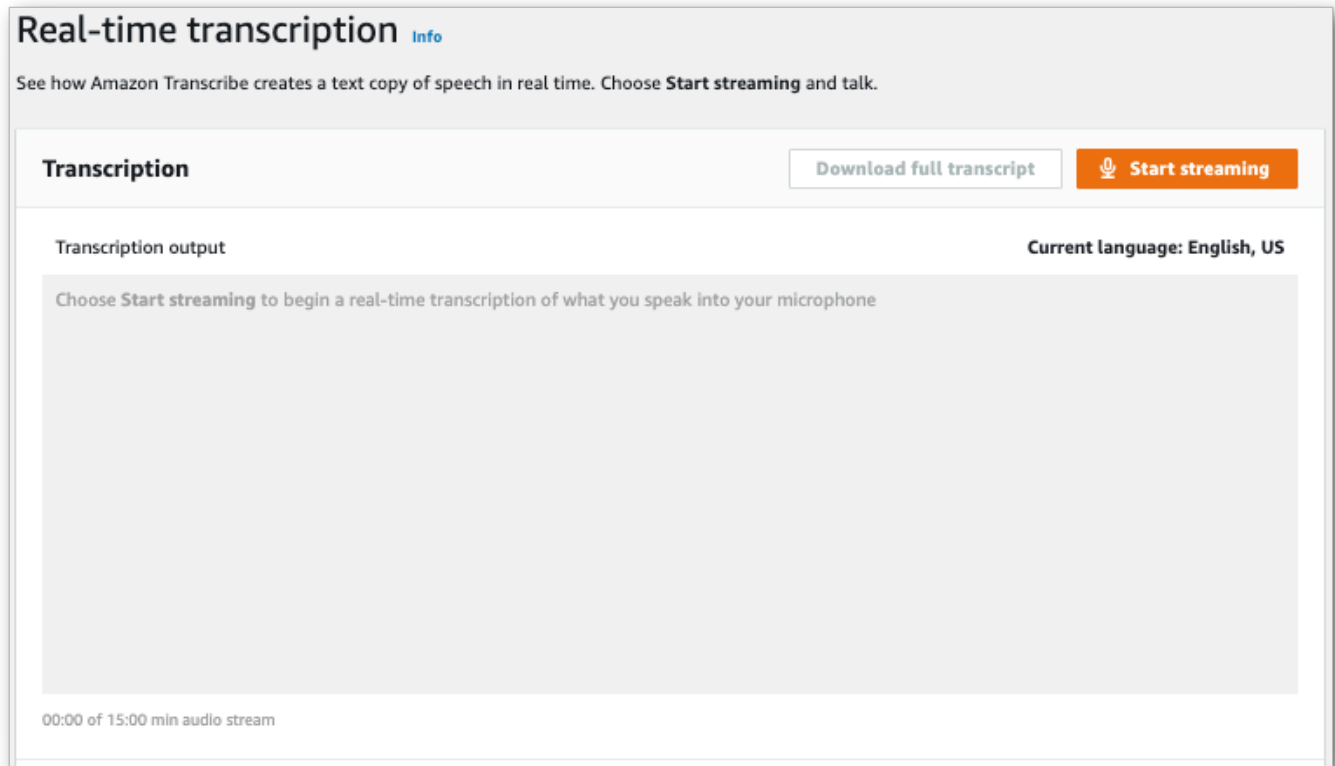
1. Em [AWS Management Console](#), selecione Transcrição em tempo real no painel de navegação esquerdo. Isso leva você à página principal do streaming do onde você pode selecionar opções.



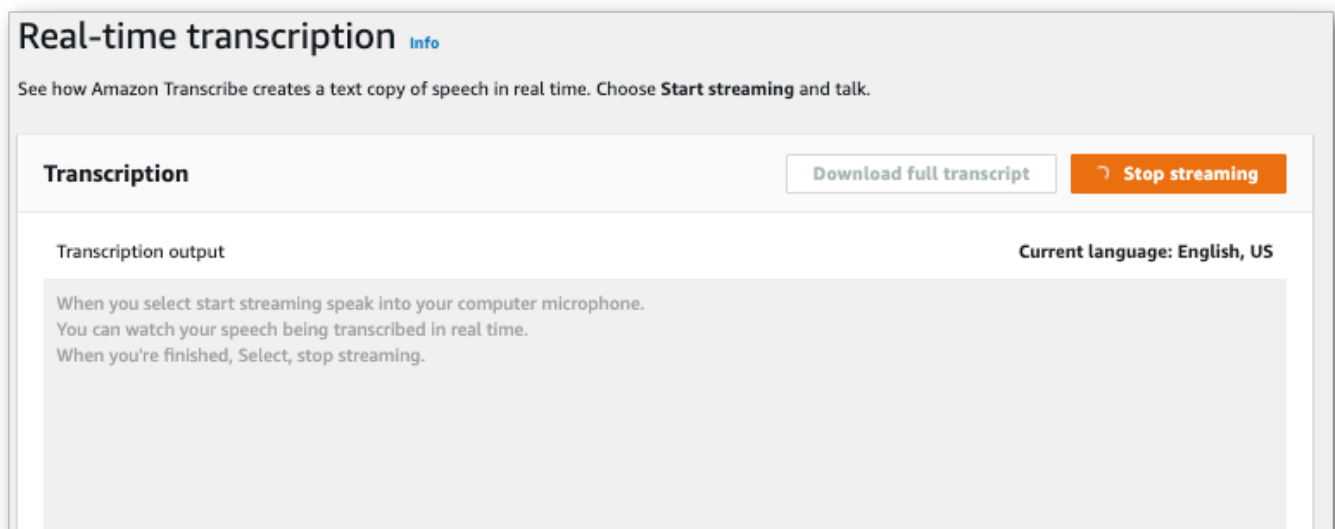
2. Abaixo da caixa de saída da transcrição, você tem a opção de selecionar várias configurações de idioma e áudio.



3. Depois de selecionar as configurações apropriadas, vá até o topo da página e escolha Iniciar transmissão e, em seguida, comece a falar no microfone do computador. Você pode ver seu discurso transcrito em tempo real.



4. Quando terminar, selecione Interromper streaming.



Agora você pode baixar sua transcrição selecionando Baixar transcrição completa.

Transcrevendo com oAWS CLI

Ao usar oAWS CLI para iniciar uma transcrição, você pode executar todos os comandos no nível da CLI. Ou você pode executar o comando que deseja usar, seguido peloRegião da AWS e pela localização de um arquivo JSON que contém o corpo da solicitação. Os exemplos deste guia mostram os dois métodos; no entanto, esta seção se concentra no método anterior.

OAWS CLI não suporta transcrições de streaming.

Antes de continuar, verifique se você tem:

- Carregou seu arquivo de mídia em umAmazon S3 bucket. Se você não tiver certeza de como criar umAmazon S3 bucket ou carregar seu arquivo, consulte [Criar seu primeiroAmazon S3 bucket](#) e [Carregar um objeto para seu bucket](#).
- Instalou [AWS CLI](#).

Você pode encontrar todos osAWS CLI comandos paraAmazon Transcribe na [Referência deAWS CLI Comandos](#).

Iniciando um novo trabalho de transcrição

Para iniciar uma nova transcrição, use ostart-transcription-job comando.

1. Em uma janela do terminal, digite o seguinte:

```
aws transcribe start-transcription-job \
```

Um '>' aparece na próxima linha e agora você pode continuar adicionando os parâmetros necessários, conforme descrito na próxima etapa.

Você também pode omitir o '\' e acrescentar todos os parâmetros, separando cada um com um espaço.

2. Com ostart-transcription-job comando, você deve incluirregiontranscription-job-name,media, e oulanguage-code ouidentify-language.

Se você quiser especificar um local de saída, incluaoutput-bucket-name em sua solicitação; se quiser especificar uma subpasta do bucket de saída especificado, inclua tambémoutput-key.

```
aws transcribe start-transcription-job \  
  --region us-west-2 \  
  --transcription-job-name my-first-transcription-job \  
  --media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
  --language-code en-US
```

Se anexar todos os parâmetros, essa solicitação terá a seguinte aparência:

```
aws transcribe start-transcription-job --region us-west-2 --transcription-job-  
name my-first-transcription-job --media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-  
input-files/my-media-file.flac --language-code en-US
```

Se você optar por não especificar um intervalo de saída usando `output-bucket-name`, Amazon Transcribe colocará sua saída de transcrição em um bucket gerenciado por serviços. As transcrições armazenadas em um bucket gerenciado por serviços expiram após 90 dias.

Amazon Transcribe responde com:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "my-first-transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "en-US",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-  
file.flac"  
    },  
    "StartTime": "2022-03-07T15:03:44.246000-08:00",  
    "CreationTime": "2022-03-07T15:03:44.229000-08:00"  
  }  
}
```

Seu trabalho de transcrição será bem-sucedido se [TranscriptionJobStatus](#) mudar de `IN_PROGRESS` para `COMPLETED`. Para ver a atualização [TranscriptionJobStatus](#), use `list-transcription-job` comando `get-transcription-job` ou, conforme mostrado na seção a seguir.

Obter o status de um trabalho de transcrição

Para obter informações sobre seu trabalho de transcrição, use o `get-transcription-job` comando.

Os únicos parâmetros necessários para esse comando são o Região da AWS local em que o trabalho está localizado e o nome do trabalho.

```
aws transcribe get-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job
```

Amazon Transcribe responde com:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "my-first-transcription-job",  
    "TranscriptionJobStatus": "COMPLETED",  
    "LanguageCode": "en-US",  
    "MediaSampleRateHertz": 48000,  
    "MediaFormat": "flac",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
    },  
    "Transcript": {  
      "TranscriptFileUri": "https://s3.the-URI-where-your-job-is-located.json"  
    },  
    "StartTime": "2022-03-07T15:03:44.246000-08:00",  
    "CreationTime": "2022-03-07T15:03:44.229000-08:00",  
    "CompletionTime": "2022-03-07T15:04:01.158000-08:00",  
    "Settings": {  
      "ChannelIdentification": false,  
      "ShowAlternatives": false  
    }  
  }  
}
```

Se você selecionou seu próprio Amazon S3 bucket para sua saída de transcrição, esse bucket está listado com `TranscriptFileUri`. Se você selecionou um bucket gerenciado por serviços, um URI temporário será fornecido; use esse URI para baixar sua transcrição.

Note

URIs temporários para Amazon S3 buckets gerenciados por serviços são válidos somente por 15 minutos. Se você receber um `AccessDenied` erro ao usar o URI, execute `get-transcription-job` solicitação novamente para obter um novo URI temporário.

Listando seus trabalhos de transcrição

Para listar todos os seus trabalhos de transcrição em um determinado Região da AWS momento, use `list-transcription-jobs` comando.

O único parâmetro necessário para esse comando é o local Região da AWS em que seus trabalhos de transcrição estão localizados.

```
aws transcribe list-transcription-jobs \  
--region us-west-2
```

Amazon Transcribe responde com:

```
{  
  "NextToken": "A-very-long-string",  
  "TranscriptionJobSummaries": [  
    {  
      "TranscriptionJobName": "my-first-transcription-job",  
      "CreationTime": "2022-03-07T15:03:44.229000-08:00",  
      "StartTime": "2022-03-07T15:03:44.246000-08:00",  
      "CompletionTime": "2022-03-07T15:04:01.158000-08:00",  
      "LanguageCode": "en-US",  
      "TranscriptionJobStatus": "COMPLETED",  
      "OutputLocationType": "SERVICE_BUCKET"  
    }  
  ]  
}
```

Excluindo seu trabalho de transcrição

Para excluir seu trabalho de transcrição, use `delete-transcription-job` comando.

Os únicos parâmetros necessários para esse comando são o Região da AWS local em que o trabalho está localizado e o nome do trabalho.


```
aws transcribe delete-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job
```

Para confirmar que sua solicitação de exclusão foi bem-sucedida, você pode executar `aws transcribe list-transcription-jobs` comando. Seu trabalho deverá aparecer na lista.

Transcrevendo com os SDKs AWS

Você pode usar SDKs para transcrições em lote e de streaming. Se você estiver transcrevendo um arquivo de mídia localizado em um Amazon S3 bucket, você está realizando uma transcrição em lote. Se estiver transcrevendo um streaming de dados de áudio em tempo real, isso significa que você está realizando uma transcrição de streaming.

Para obter uma lista das linguagens de programação com as quais você pode usar Amazon Transcribe, consulte [Linguagens de programação compatíveis](#). Observe que as transcrições de streaming não são compatíveis com todos os AWS SDKs. Para ver os formatos de mídia aceitos e outros requisitos e restrições de mídia, consulte [Entrada e saída de dados](#).

Para obter mais informações sobre todos os AWS SDKs e ferramentas de criação disponíveis, consulte [Ferramentas para desenvolver](#). AWS

Tip

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

Você também pode encontrar exemplos de código do SDK nesses GitHub repositórios:

- [AWS Exemplos de código](#)
- [Amazon Transcribe Exemplos](#)

Transcrições em lote

Você pode criar transcrições em lote usando o URI de um arquivo de mídia localizado em um bucket do Amazon S3 . Se você não tiver certeza de como criar um Amazon S3 bucket ou fazer upload do seu arquivo, consulte [Criar seu primeiro bucket do S3](#) e [fazer upload de um objeto no seu bucket](#).

Java

```
import software.amazon.awssdk.auth.credentials.AwsCredentialsProvider;
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.transcribe.TranscribeClient;
import software.amazon.awssdk.services.transcribe.model.*;
import software.amazon.awssdk.services.transcribestreaming.model.LanguageCode;

public class TranscribeDemoApp {
    private static final Region REGION = Region.US_WEST_2;
    private static TranscribeClient client;

    public static void main(String args[]) {

        client = TranscribeClient.builder()
            .credentialsProvider(getCredentials())
            .region(REGION)
            .build();

        String transcriptionJobName = "my-first-transcription-job";
        String mediaType = "flac"; // can be other types
        Media myMedia = Media.builder()
            .mediaFileUri("s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-
file.flac")
            .build();

        String outputS3BucketName = "s3://DOC-EXAMPLE-BUCKET";
        // Create the transcription job request
        StartTranscriptionJobRequest request =
StartTranscriptionJobRequest.builder()
            .transcriptionJobName(transcriptionJobName)
            .languageCode(LanguageCode.EN_US.toString())
            .mediaSampleRateHertz(16000)
            .mediaFormat(mediaType)
            .media(myMedia)
            .outputBucketName(outputS3BucketName)
            .build();

        // send the request to start the transcription job
        StartTranscriptionJobResponse startJobResponse =
client.startTranscriptionJob(request);

        System.out.println("Created the transcription job");
    }
}
```

```
        System.out.println(startJobResponse.transcriptionJob());

        // Create the get job request
        GetTranscriptionJobRequest getJobRequest =
GetTranscriptionJobRequest.builder()
            .transcriptionJobName(transcriptionJobName)
            .build();

        // send the request to get the transcription job including the job status
        GetTranscriptionJobResponse getJobResponse =
client.getTranscriptionJob(getJobRequest);

        System.out.println("Get the transcription job request");
        System.out.println(getJobResponse.transcriptionJob());
    }

    private static AwsCredentialsProvider getCredentials() {
        return DefaultCredentialsProvider.create();
    }
}
}
```

JavaScript

```
const { TranscribeClient, StartTranscriptionJobCommand } = require("@aws-sdk/client-transcribe"); // CommonJS import

const region = "us-west-2";
const credentials = {
    "accessKeyId": "",
    "secretAccessKey": "",
};

const input = {
    TranscriptionJobName: "my-first-transcription-job",
    LanguageCode: "en-US",
    Media: {
        MediaFileUri: "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
    },
    OutputBucketName: "DOC-EXAMPLE-BUCKET",
};

async function startTranscriptionRequest() {
```

```

const transcribeConfig = {
  region,
  credentials
};
const transcribeClient = new TranscribeClient(transcribeConfig);
const transcribeCommand = new StartTranscriptionJobCommand(input);
try {
  const transcribeResponse = await transcribeClient.send(transcribeCommand);
  console.log("Transcription job created, the details:");
  console.log(transcribeResponse.TranscriptionJob);
} catch(err) {
  console.log(err);
}
}

startTranscriptionRequest();

```

Python

```

import time
import boto3

def transcribe_file(job_name, file_uri, transcribe_client):
    transcribe_client.start_transcription_job(
        TranscriptionJobName = job_name,
        Media = {
            'MediaFileUri': file_uri
        },
        MediaFormat = 'flac',
        LanguageCode = 'en-US'
    )

    max_tries = 60
    while max_tries > 0:
        max_tries -= 1
        job = transcribe_client.get_transcription_job(TranscriptionJobName =
job_name)
        job_status = job['TranscriptionJob']['TranscriptionJobStatus']
        if job_status in ['COMPLETED', 'FAILED']:
            print(f"Job {job_name} is {job_status}.")
            if job_status == 'COMPLETED':
                print(
                    f"Download the transcript from\n"

```

```
        f"\t{job['TranscriptionJob']['Transcript']
['TranscriptFileUri']}."))
        break
    else:
        print(f"Waiting for {job_name}. Current status is {job_status}.")
        time.sleep(10)

def main():
    transcribe_client = boto3.client('transcribe', region_name = 'us-west-2')
    file_uri = 's3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac'
    transcribe_file('Example-job', file_uri, transcribe_client)

if __name__ == '__main__':
    main()
```

Transcrições de streaming

É possível criar transcrições de streaming usando um arquivo de mídia transmitido ou um streaming de mídia ao vivo.

Observe que o padrão não AWS SDK for Python (Boto3) é compatível com Amazon Transcribe streaming. Para iniciar uma transcrição de streaming usando Python, use esse SDK [assíncrono](#) do Python para Amazon Transcribe

Java

O exemplo a seguir é um programa Java que transcreve áudio de streaming.

Para executar esse exemplo, observe o seguinte:

- Você deve usar o [AWS SDK para Java 2.x](#).
- Os clientes devem usar o Java 1.8 para serem compatíveis com o [AWS SDK para Java 2.x](#).
- A taxa de amostragem especificada deve corresponder à taxa de amostragem real do fluxo de áudio.

Veja também: [Tente novamente o cliente para Amazon Transcribe streaming \(Java SDK\)](#). Esse código gerencia a conexão com o Amazon Transcribe e tenta enviar dados novamente quando há

erros na conexão. Por exemplo, se houver um erro temporário na rede, esse cliente reenviará a solicitação que falhou.

```
public class TranscribeStreamingDemoApp {
    private static final Region REGION = Region.US_WEST_2;
    private static TranscribeStreamingAsyncClient client;

    public static void main(String args[]) throws URISyntaxException,
        ExecutionException, InterruptedException, LineUnavailableException {

        client = TranscribeStreamingAsyncClient.builder()
            .credentialsProvider(getCredentials())
            .region(REGION)
            .build();

        CompletableFuture<Void> result =
client.startStreamTranscription(getRequest(16_000),
    new AudioStreamPublisher(getStreamFromMic()),
    getResponseHandler());

        result.get();
        client.close();
    }

    private static InputStream getStreamFromMic() throws LineUnavailableException {

        // Signed PCM AudioFormat with 16,000 Hz, 16 bit sample size, mono
        int sampleRate = 16000;
        AudioFormat format = new AudioFormat(sampleRate, 16, 1, true, false);
        DataLine.Info info = new DataLine.Info(TargetDataLine.class, format);

        if (!AudioSystem.isLineSupported(info)) {
            System.out.println("Line not supported");
            System.exit(0);
        }

        TargetDataLine line = (TargetDataLine) AudioSystem.getLine(info);
        line.open(format);
        line.start();

        InputStream audioStream = new AudioInputStream(line);
        return audioStream;
    }
}
```

```
private static AwsCredentialsProvider getCredentials() {
    return DefaultCredentialsProvider.create();
}

private static StartStreamTranscriptionRequest getRequest(Integer
mediaSampleRateHertz) {
    return StartStreamTranscriptionRequest.builder()
        .languageCode(LanguageCode.EN_US.toString())
        .mediaEncoding(MediaEncoding.PCM)
        .mediaSampleRateHertz(mediaSampleRateHertz)
        .build();
}

private static StartStreamTranscriptionResponseHandler getResponseHandler() {
    return StartStreamTranscriptionResponseHandler.builder()
        .onResponse(r -> {
            System.out.println("Received Initial response");
        })
        .onError(e -> {
            System.out.println(e.getMessage());
            StringWriter sw = new StringWriter();
            e.printStackTrace(new PrintWriter(sw));
            System.out.println("Error Occurred: " + sw.toString());
        })
        .onComplete(() -> {
            System.out.println("=== All records stream successfully ===");
        })
        .subscriber(event -> {
            List<Result> results = ((TranscriptEvent)
event).transcript().results();
            if (results.size() > 0) {
                if (!
results.get(0).alternatives().get(0).transcript().isEmpty()) {

System.out.println(results.get(0).alternatives().get(0).transcript());
                }
            }
        })
        .build();
}

private InputStream getStreamFromFile(String myMediaFileName) {
    try {
```

```
        File inputFile = new
File(getClass().getClassLoader().getResource(myMediaFileName).getFile());
        InputStream audioStream = new FileInputStream(inputFile);
        return audioStream;
    } catch (FileNotFoundException e) {
        throw new RuntimeException(e);
    }
}

private static class AudioStreamPublisher implements Publisher<AudioStream> {
    private final InputStream inputStream;
    private static Subscription currentSubscription;

    private AudioStreamPublisher(InputStream inputStream) {
        this.inputStream = inputStream;
    }

    @Override
    public void subscribe(Subscriber<? super AudioStream> s) {

        if (this.currentSubscription == null) {
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        } else {
            this.currentSubscription.cancel();
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        }
        s.onSubscribe(currentSubscription);
    }
}

public static class SubscriptionImpl implements Subscription {
    private static final int CHUNK_SIZE_IN_BYTES = 1024 * 1;
    private final Subscriber<? super AudioStream> subscriber;
    private final InputStream inputStream;
    private ExecutorService executor = Executors.newFixedThreadPool(1);
    private AtomicLong demand = new AtomicLong(0);

    SubscriptionImpl(Subscriber<? super AudioStream> s, InputStream inputStream)
{
        this.subscriber = s;
        this.inputStream = inputStream;
    }
}
```



```
@Override
public void request(long n) {
    if (n <= 0) {
        subscriber.onError(new IllegalArgumentException("Demand must be
positive"));
    }

    demand.getAndAdd(n);

    executor.submit(() -> {
        try {
            do {
                ByteBuffer audioBuffer = getNextEvent();
                if (audioBuffer.remaining() > 0) {
                    AudioEvent audioEvent =
audioEventFromBuffer(audioBuffer);
                    subscriber.onNext(audioEvent);
                } else {
                    subscriber.onComplete();
                    break;
                }
            } while (demand.decrementAndGet() > 0);
        } catch (Exception e) {
            subscriber.onError(e);
        }
    });
}

@Override
public void cancel() {
    executor.shutdown();
}

private ByteBuffer getNextEvent() {
    ByteBuffer audioBuffer = null;
    byte[] audioBytes = new byte[CHUNK_SIZE_IN_BYTES];

    int len = 0;
    try {
        len = inputStream.read(audioBytes);

        if (len <= 0) {
            audioBuffer = ByteBuffer.allocate(0);
        } else {
```

```

        audioBuffer = ByteBuffer.wrap(audioBytes, 0, len);
    }
} catch (IOException e) {
    throw new UncheckedIOException(e);
}

return audioBuffer;
}

private AudioEvent audioEventFromBuffer(ByteBuffer bb) {
    return AudioEvent.builder()
        .audioChunk(SdkBytes.fromByteBuffer(bb))
        .build();
}
}
}
}

```

JavaScript

```

const {
    TranscribeStreamingClient,
    StartStreamTranscriptionCommand,
} = require("@aws-sdk/client-transcribe-streaming");
const { createReadStream } = require("fs");
const { join } = require("path");

const audio = createReadStream(join(__dirname, "my-media-file.flac"),
    { highWaterMark: 1024 * 16});

const LanguageCode = "en-US";
const MediaEncoding = "pcm";
const MediaSampleRateHertz = "16000";
const credentials = {
    "accessKeyId": "",
    "secretAccessKey": "",
};
};

async function startRequest() {
    const client = new TranscribeStreamingClient({
        region: "us-west-2",
        credentials
    });

    const params = {

```

```
    LanguageCode,  
    MediaEncoding,  
    MediaSampleRateHertz,  
    AudioStream: (async function* () {  
      for await (const chunk of audio) {  
        yield {AudioEvent: {AudioChunk: chunk}};  
      }  
    })(),  
  });  
const command = new StartStreamTranscriptionCommand(params);  
// Send transcription request  
const response = await client.send(command);  
// Start to print response  
try {  
  for await (const event of response.TranscriptResultStream) {  
    console.log(JSON.stringify(event));  
  }  
} catch(err) {  
  console.log("error")  
  console.log(err)  
}  
}  
startRequest();
```

Python

O exemplo a seguir é um programa Python que transcreve áudio de streaming.

Para executar esse exemplo, observe o seguinte:

- Você deve usar este [SDK para Python](#).
- A taxa de amostragem especificada deve corresponder à taxa de amostragem real do fluxo de áudio.

```
import asyncio  
# This example uses aiofile for asynchronous file reads.  
# It's not a dependency of the project but can be installed  
# with `pip install aiofile`.  
import aiofile  
  
from amazon_transcribe.client import TranscribeStreamingClient  
from amazon_transcribe.handlers import TranscriptResultStreamHandler
```

```
from amazon_transcribe.model import TranscriptEvent

"""
Here's an example of a custom event handler you can extend to
process the returned transcription results as needed. This
handler will simply print the text out to your interpreter.
"""

class MyEventHandler(TranscriptResultStreamHandler):
    async def handle_transcript_event(self, transcript_event: TranscriptEvent):
        # This handler can be implemented to handle transcriptions as needed.
        # Here's an example to get started.
        results = transcript_event.transcript.results
        for result in results:
            for alt in result.alternatives:
                print(alt.transcript)

async def basic_transcribe():
    # Set up our client with your chosen Region
    client = TranscribeStreamingClient(region = "us-west-2")

    # Start transcription to generate async stream
    stream = await client.start_stream_transcription(
        language_code = "en-US",
        media_sample_rate_hz = 16000,
        media_encoding = "pcm",
    )

    async def write_chunks():
        # NOTE: For pre-recorded files longer than 5 minutes, the sent audio
        # chunks should be rate limited to match the real-time bitrate of the
        # audio stream to avoid signing issues.
        async with aiofile.AIOFile('filepath/my-media-file.flac', 'rb') as afp:
            reader = aiofile.Reader(afp, chunk_size = 1024 * 16)
            async for chunk in reader:
                await stream.input_stream.send_audio_event(audio_chunk = chunk)
            await stream.input_stream.end_stream()

    # Instantiate our handler and start processing events
    handler = MyEventHandler(stream.output_stream)
    await asyncio.gather(write_chunks(), handler.handle_events())

loop = asyncio.get_event_loop()
loop.run_until_complete(basic_transcribe())
```

```
loop.close()
```

C++

Consulte no capítulo “Exemplos de código” [o exemplo do SDK de streaming para C++](#).

Usando esse serviço com um AWS SDK

AWS kits de desenvolvimento de software (SDKs) estão disponíveis para muitas linguagens de programação populares. Cada SDK fornece uma API, exemplos de código e documentação que facilitam a criação de aplicações em seu idioma preferido pelos desenvolvedores.

Documentação do SDK	Exemplos de código
AWS SDK for C++	AWS SDK for C++ exemplos de código
AWS CLI	AWS CLI exemplos de código
AWS SDK for Go	AWS SDK for Go exemplos de código
AWS SDK for Java	AWS SDK for Java exemplos de código
AWS SDK for JavaScript	AWS SDK for JavaScript exemplos de código
AWS SDK para Kotlin	AWS SDK para Kotlin exemplos de código
AWS SDK for .NET	AWS SDK for .NET exemplos de código
AWS SDK for PHP	AWS SDK for PHP exemplos de código
AWS Tools for PowerShell	Ferramentas para exemplos PowerShell de código
AWS SDK for Python (Boto3)	AWS SDK for Python (Boto3) exemplos de código
AWS SDK for Ruby	AWS SDK for Ruby exemplos de código
AWS SDK para Rust	AWS SDK para Rust exemplos de código

Documentação do SDK	Exemplos de código
SDK da AWS para SAP ABAP	SDK da AWS para SAP ABAP exemplos de código
AWS SDK for Swift	AWS SDK for Swift exemplos de código

Para obter exemplos específicos deste serviço, consulte [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#).

Exemplo de disponibilidade

Você não consegue encontrar o que precisa? Solicite um código de exemplo no link Fornecer feedback na parte inferior desta página.

Transcrevendo com HTTP ou WebSockets

Amazon Transcribe suporta HTTP para transcrições em lote (HTTP/1.1) e streaming (HTTP/2). WebSockets são compatíveis com transcrições de streaming.

Se você estiver transcrevendo um arquivo de mídia localizado em um Amazon S3 bucket, você está realizando uma transcrição em lote. Se você estiver transcrevendo um fluxo de dados de áudio em tempo real, você está realizando uma transcrição de streaming.

Tanto o HTTP quanto o HTTP WebSockets exigem que você autentique sua solicitação usando cabeçalhos AWS Signature Versão 4. Consulte [Solicitações AWS da API de assinatura](#) para obter mais informações.

Transcrições em Batch

Você pode fazer uma solicitação HTTP em lote usando os seguintes cabeçalhos:

- host
- x-amz-target
- tipo de conteúdo
- x-amz-content-sha256

- x-amz-date
- autorização

Veja um exemplo de uma crie um `startTranscriptionJob` pedido:

```
POST /transcribe HTTP/1.1
host: transcribe.us-west-2.amazonaws.com
x-amz-target: com.amazonaws.transcribe.Transcribe.StartTranscriptionJob
content-type: application/x-amz-json-1.1
x-amz-content-sha256: string
x-amz-date: YYYYMMDDTHHMMSSZ
authorization: AWS4-HMAC-SHA256 Credential=access-key/YYYYMMSS/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string


{
  "TranscriptionJobName": "my-first-transcription-job",
  "LanguageCode": "en-US",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/"
}
```

Operações e parâmetros adicionais estão listados na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#). Outros elementos de assinatura são detalhados em [Elementos de uma solicitação de AWS assinatura versão 4](#).

Transcrições de streaming

Transmita transcrições usando HTTP/2 e WebSockets envolve mais do que o uso de SDKs. Recomendamos revisar a [Configurar uma transcrição de streaming](#) seção antes de configurar sua primeira transmissão.

Para obter mais informações sobre esses métodos, consulte [Configurar um fluxo de HTTP/2](#) ou [Configurando um WebSocket stream](#).

 Note

É altamente recomendável usar um SDK para transcrições de streaming. Para obter uma lista dos SDKs compatíveis, consulte [Linguagens de programação compatíveis](#).

Transcrever streaming de áudio

Usando o Amazon Transcribe streaming, você pode produzir transcrições em tempo real para seu conteúdo de mídia. Ao contrário das transcrições em lote, que envolvem o upload de arquivos de mídia, a mídia de streaming é entregue Amazon Transcribe em tempo real. Amazon Transcribe em seguida, retorna uma transcrição, também em tempo real.

O streaming pode incluir mídia pré-gravada (filmes, músicas e podcasts) e mídia em tempo real (transmissões de notícias ao vivo). Os casos de uso comuns de streaming Amazon Transcribe incluem legendas ocultas ao vivo para eventos esportivos e monitoramento em tempo real do áudio da central de atendimento.

O conteúdo de streaming é fornecido como uma série de pacotes de dados sequenciais ou “blocos” que o Amazon Transcribe transcreve instantaneamente. As vantagens de usar streaming em lote incluem speech-to-text recursos em tempo real em seus aplicativos e tempos de transcrição mais rápidos. No entanto, essa maior velocidade pode ter limitações de precisão em alguns casos.

Amazon Transcribe oferece as seguintes opções para streaming:

- [SDKs](#) (preferencial)
- [HTTP/2](#)
- [WebSockets](#)
- [AWS Management Console](#)

Para transcrever o streaming de áudio no AWS Management Console, fale no microfone do computador.

Tip

Para exemplos de código do SDK, consulte o [repositório AWS Samples](#) em GitHub

Os formatos de áudio compatíveis para transcrições de streaming são:

- FLAC
- Áudio codificado em OPUS em um contêiner Ogg
- PCM (somente formatos de áudio little-endian assinados de 16 bits, que não incluem WAV)

Formatos sem perdas (FLAC ou PCM) são recomendados.

Note

As transcrições de streaming não são compatíveis com todos os idiomas. Consulte a coluna “Entrada de dados” na [tabela de idiomas oferecidos](#) para obter detalhes.

Para ver a disponibilidade da Amazon Transcribe região para transcrições de streaming, consulte: [Amazon Transcribe Endpoints](#) e cotas.

Práticas recomendadas

As recomendações a seguir melhoram a eficiência da transcrição de streaming:

- Se possível, use áudio codificado em PCM.
- Garanta que o streaming seja o mais próximo possível do tempo real.
- A latência depende do tamanho dos fragmentos de áudio. Se você conseguir especificar o tamanho do fragmento com o tipo de áudio (como com o PCM), defina cada fragmento entre 50 ms e 200 ms. É possível calcular o tamanho do fragmento de áudio usando a seguinte fórmula:

```
chunk_size_in_bytes = chunk_duration_in_millisecond / 1000 * audio_sample_rate * 2
```

- Use um tamanho de fragmento uniforme.
- Especifique corretamente o número de canais de áudio.
- Com o áudio PCM de canal único, cada amostra consiste em 2 bytes, portanto, cada bloco deve consistir em um número par de bytes.
- Com o áudio PCM de canal duplo, cada amostra consiste em 4 bytes, portanto, cada bloco deve ser um múltiplo de 4 bytes.
- Quando o streaming de áudio não contiver fala, codifique e envie a mesma quantidade de silêncio. Por exemplo, o silêncio para PCM é um streaming de zero bytes.
- Especifique a taxa de amostragem correta para o áudio. Se possível, grave com uma taxa de amostragem de 16.000 Hz, que fornece a melhor conciliação entre qualidade e volume de dados enviados pela rede. Observe que a maioria dos microfones de última geração grava a 44.100 Hz ou 48.000 Hz.

Streaming e resultados parciais

Como o streaming funciona em tempo real, as transcrições são produzidas em resultados parciais. Amazon Transcribe interrompe o fluxo de áudio de entrada com base em segmentos naturais da fala, como uma mudança no alto-falante ou uma pausa no áudio. A transcrição é retornada à aplicação em um fluxo de eventos de transcrição, em que cada resposta contém mais fala transcrita à medida que o segmento inteiro é transcrito.

Um bloco de código semelhante é mostrado a seguir. Você pode ver esse processo na prática fazendo login no [AWS Management Console](#), selecionando Transcrição em tempo real e falando pelo microfone. Observe o painel Saída da transcrição enquanto você fala.

Nesse exemplo, cada linha é o resultado parcial de um segmento de áudio.

```
The
The Amazon.
The Amazon is
The Amazon is the law.
The Amazon is the largest
The Amazon is the largest ray
The Amazon is the largest rain for
The Amazon is the largest rainforest.
The Amazon is the largest rainforest on the
The Amazon is the largest rainforest on the planet.
```

Esses resultados parciais estão presentes na saída da transcrição dentro dos objetos [Results](#). Também nesse bloco de objetos há um `IsPartial` campo. Se esse campo for verdadeiro, o segmento de transcrição ainda não estará completo. Você pode ver a diferença entre um segmento incompleto e um completo abaixo:

```
"IsPartial": true (incomplete segment)

"Transcript": "The Amazon is the largest rainforest."

"EndTime": 4.545,
"IsPartial": true,
"ResultId": "12345a67-8bc9-0de1-2f34-a5b678c90d12",
"StartTime": 0.025

"IsPartial": false (complete segment)
```

```
"Transcript": "The Amazon is the largest rainforest on the planet."  
  
"EndTime": 6.025,  
"IsPartial": false,  
"ResultId": "34567e89-0fa1-2bc3-4d56-78e90123456f",  
"StartTime": 0.025
```

Cada palavra em um segmento completo tem uma pontuação de confiança correspondente, que é um valor entre 0 e 1. Um valor maior indica uma probabilidade de a palavra ser transcrita corretamente.

Tip

Em um segmento de áudio, `StartTime` e `EndTime` podem ser usados para sincronizar a saída da transcrição com o diálogo de vídeo.

Se você estiver executando uma aplicação que exija baixa latência, é aconselhável usar a [estabilização de resultados parciais](#).

Estabilização de resultados parciais

Amazon Transcribe começa a retornar os resultados da transcrição assim que você começa a transmitir seu áudio. Ele retorna esses resultados parciais de forma incremental até gerar um resultado final no nível de um segmento de fala natural. Um segmento de fala natural é uma fala contínua que contém uma pausa ou uma mudança de locutor.

Amazon Transcribe continua produzindo resultados parciais até gerar o resultado final da transcrição para um segmento de fala. Como o reconhecimento de fala pode revisar as palavras à medida que ganha mais contexto, as transcrições de streaming podem mudar um pouco a cada nova saída de resultados parciais.

Esse processo oferece duas opções para cada segmento de fala:

- Aguardar o segmento finalizado
- Usar os resultados parciais do segmento

A estabilização parcial do resultado muda a forma como Amazon Transcribe produz o resultado final da transcrição para cada segmento completo. Quando ativada, somente as últimas palavras

dos resultados parciais podem mudar. Por causa disso, a precisão da transcrição pode ser afetada. No entanto, a transcrição é retornada mais rapidamente do que sem a estabilização dos resultados parciais. Essa redução na latência pode ser benéfica ao legendar vídeos ou gerar legendas para streamings ao vivo.

Os exemplos a seguir mostram como o mesmo streaming de áudio é tratado quando a estabilização de resultados parciais não está ativada e quando está. Observe que você pode definir o nível de estabilidade como baixo, médio ou alto. A baixa estabilidade fornece a mais alta precisão. A transcrição de alta estabilidade é mais rápida, mas oferece uma precisão um pouco menor.

"Transcript":	"EndTime":	"IsPartial":
Estabilização de resultados parciais não habilitada		
<pre>The The The Amazon. The Amazon is The Amazon is the law. The Amazon is the largest The Amazon is the largest ray The Amazon is the largest rain for The Amazon is the largest rainforest. The Amazon is the largest rainforest on the The Amazon is the largest rainforest on the planet. The Amazon is the largest rainforest on the planet. The Amazon is the largest rainforest on the planet.</pre>	<pre>0.545 1.045 1.545 2.045 2.545 3.045 3.545 4.045 4.545 5.045 5.545 6.025 6.025</pre>	<pre>true true true true true true true true true true true true true false</pre>

Estabilização de resultados parciais habilitada (alta estabilidade)

"Transcript":	"EndTime":	"IsPartial":
The	0.515	true
The	1.015	true
The Amazon.	1.515	true
The Amazon is	2.015	true
The Amazon is the large	2.515	true
The Amazon is the	3.015	true
largest	3.515	true
The Amazon is the	4.015	true
largest rainfall.	4.515	true
The Amazon is the	5.015	true
largest rain forest.	5.515	true
The Amazon is the	6.015	true
largest rain forest on	6.335	true
The Amazon is the	6.335	false
largest rain forest on		
the planet.		
The Amazon is the		
largest rain forest on		
the planet.		
The Amazon is the		
largest rain forest on		
the planet.		
The Amazon is the		
largest rain forest on		
the planet.		
The Amazon is the		
largest rain forest on		
the planet.		

Quando você ativa a estabilização parcial do resultado, Amazon Transcribe usa um `Stable` campo para indicar se um item é estável, onde "item" se refere a uma palavra transcrita ou sinal de pontuação. Os valores para `Stable` são `true` ou `false`. Os itens marcados como `false` (não estáveis) têm maior probabilidade de mudar à medida que o segmento é transcrito. Entretanto, os itens marcados como `true` (estável) não serão alterados.

Você pode optar por renderizar palavras não estáveis para que as legendas se alinhem à fala. Mesmo que as legendas mudem um pouco à medida que o contexto é adicionado, essa é uma

experiência de usuário melhor do que intermitências periódicas de texto, que podem ou não se alinhar à fala.

Você também pode optar por exibir palavras não estáveis em um formato diferente, como itálico, para indicar aos espectadores que essas palavras podem mudar. A exibição de resultados parciais limita a quantidade de texto exibida em determinado momento. Isso pode ser importante quando você lida com restrições de espaço, como acontece com legendas de vídeo.

Mergulhe mais fundo com o Blog AWS de Machine Learning

Para saber mais sobre como melhorar a precisão com transcrições em tempo real, consulte:

- [Melhore a experiência de transcrição de streaming com estabilização Amazon Transcribe parcial dos resultados](#)
- [“What was that?” Increasing subtitle accuracy for live broadcasts using Amazon Transcribe](#)

Exemplo de saída de estabilização de resultados parciais

O exemplo de saída a seguir mostra sinalizadores `Stable` para um segmento incompleto (`"IsPartial": true`). Você pode ver que as palavras “to” e “Amazon” não são estáveis e, portanto, podem mudar antes que o segmento seja finalizado.

```
"Transcript": {
  "Results": [
    {
      "Alternatives": [
        {
          "Items": [
            {
              "Content": "Welcome",
              "EndTime": 2.4225,
              "Stable": true,
              "StartTime": 1.65,
              "Type": "pronunciation",
              "VocabularyFilterMatch": false
            },
            {
              "Content": "to",
              "EndTime": 2.8325,
              "Stable": false,
```

```
        "StartTime": 2.4225,
        "Type": "pronunciation",
        "VocabularyFilterMatch": false
    },
    {
        "Content": "Amazon",
        "EndTime": 3.635,
        "Stable": false,
        "StartTime": 2.8325,
        "Type": "pronunciation",
        "VocabularyFilterMatch": false
    },
    {
        "Content": ".",
        "EndTime": 3.635,
        "Stable": false,
        "StartTime": 3.635,
        "Type": "punctuation",
        "VocabularyFilterMatch": false
    }
    ],
    "Transcript": "Welcome to Amazon."
}
],
"EndTime": 4.165,
"IsPartial": true,
"ResultId": "12345a67-8bc9-0de1-2f34-a5b678c90d12",
"StartTime": 1.65
}
]
```

Configurar uma transcrição de streaming

Esta seção expande a seção principal de [streaming](#). O objetivo é fornecer informações para usuários que desejam configurar seu stream com HTTP/2 ou WebSockets diretamente, em vez de com um AWS SDK. As informações nesta seção também podem ser usadas para criar seu próprio SDK.

⚠ Important

É altamente recomendável usar SDKs em vez de usar HTTP/2 e diretamente. WebSockets Os SDKs são o método mais simples e confiável para transcrever fluxos de dados. Para começar a transmitir usando um AWS SDK, consulte [Transcrevendo com os SDKs AWS](#).

Configurar um fluxo de HTTP/2

Os principais componentes de um [protocolo HTTP/2](#) para streaming de solicitações de transcrição são: Amazon Transcribe

- Um quadro de cabeçalho. Ele contém os cabeçalhos HTTP/2 para sua solicitação e uma assinatura no cabeçalho de autorização que é Amazon Transcribe usada como assinatura inicial para assinar os quadros de dados.
- Um ou mais quadros de mensagens na [codificação de fluxo de eventos](#) que contém metadados e bytes de áudio bruto.
- Um quadro final. Essa é uma mensagem assinada na [codificação de fluxo de eventos](#) com um corpo vazio.

i Note

Amazon Transcribe suporta apenas um stream por sessão HTTP/2. Se você tentar usar vários fluxos, sua solicitação de transcrição falhará.

1. Anexe a política a seguir à IAM função que faz a solicitação. Consulte [Adicionar IAM políticas](#) para obter mais informações.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "my-transcribe-http2-policy",
      "Effect": "Allow",
      "Action": "transcribe:StartStreamTranscription",
      "Resource": "*"
    }
  ]
}
```

```
]
}
```

2. Para iniciar a sessão, envie uma solicitação HTTP/2 para o Amazon Transcribe.

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: YYYYMMDDTHHMMSSZ
Authorization: AWS4-HMAC-SHA256 Credential=access-key/YYYYMMDD/us-west-2/
transcribe/aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-
amz-date;x-amz-target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
transfer-encoding: chunked
```

Operações e parâmetros adicionais estão listados na [Referência da API](#). Os parâmetros comuns a todas as operações de API da AWS estão listados na seção [Common Parameters](#).

Amazon Transcribe envia a seguinte resposta:

```
HTTP/2.0 200
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-request-id: 8a08df7d-5998-48bf-a303-484355b4ab4e
x-amzn-transcribe-session-id: b4526fcf-5eee-4361-8192-d1cb9e9d6887
content-type: application/json
```

3. Crie um evento de áudio que contenha os dados de áudio. Combine os cabeçalhos descritos na tabela a seguir com um bloco de bytes de áudio em uma mensagem codificada por evento. Para criar a carga para a mensagem do evento, use um buffer no formato de byte bruto.

Comprimento em bytes do nome do cabeçalho	Nome do cabeçalho (string)	Tipo de valor de cabeçalho	Comprimento em bytes da string do valor	String do valor (UTF-8)
13	:content-type	7	24	application/octet-stream
11	:event-type	7	10	AudioEvent
13	:message-type	7	5	evento

Os dados binários nessa solicitação de exemplo são codificados em base64. Em uma solicitação real, os dados são bytes brutos.

```
:content-type: "application/vnd.amazon.eventstream"
:event-type: "AudioEvent"
:message-type: "event"
Uk1GRjzxPQBXQVZFM10IBAAAAABAAEAgD4AAAB9AAACABAAZGF0YVVTwPQAAAAAAAAAAAAAAAAAAD//wIA/
f8EAA==
```

4. Crie uma mensagem de áudio que contenha seus dados de áudio.
 - a. O quadro de dados da mensagem de áudio contém cabeçalhos de codificação de eventos que incluem a data atual e uma assinatura para o bloco de áudio e o evento de áudio.

Comprimento em bytes do nome do cabeçalho	Nome do cabeçalho (string)	Tipo de valor de cabeçalho	Comprimento em bytes da string do valor	Valor
16	:chunk-signature	6	varia	assinatura gerada
5	:date	8	8	timestamp

Os dados binários nessa solicitação são codificados em base64. Em uma solicitação real, os dados são bytes brutos.

```
:date: 2019-01-29T01:56:17.291Z
:chunk-signature: signature

AAAA0gAAAIKVoRFcTTcjb250ZW50LXR5cGUHABhhcHBsaWNhdGlvbi9vY3RldC1zdHJlYW0L0mV2ZW50LXR5
cGUHAAPBdWRpb0V2ZW50DTptZXNzYwdlLXR5cGUHAAV1dmVudAxDb256ZW50LVR5cGUHABphcHBsaWNhdGlv
bi94LWFtei1qc29uLFEuMVJJRkY88T0AV0FWRWZtdCAQAAAAAQAABAIA
+AAAAfQAAAQAQAGRhdGFU8D0AAAAA
AAAAAAAAAAAA//8CAP3/BAC7QLFf
```

- b. Construa uma string para assinar, conforme descrito em [Criar uma string para assinar](#) para o for Signature Version 4. A string segue este formato:

```
String stringToSign =
"AWS4-HMAC-SHA256" +
"\n" +
DateTime +
"\n" +
Keypath +
"\n" +
Hex(priorSignature) +
"\n" +
HexHash(nonSignatureHeaders) +
"\n" +
HexHash(payload);
```

- *DateTime*: a data e a hora em que a assinatura foi criada. O formato é YYYYMMDDTHHMMSSZ, onde YYYY = ano, mm = mês, DD = dia, HH = hora, MM = minuto, SS = segundos e “T” e “Z” são caracteres fixos. Para obter mais informações, consulte [Handling Dates in Signature Version 4](#).
- *Keypath*: o escopo da assinatura no formato date/region/service/aws4_request. Por exemplo, 20220127/us-west-2/transcribe/aws4_request.
- *Hex*: uma função que codifica a entrada em uma representação hexadecimal.
- *priorSignature*: a assinatura do quadro anterior. Para o primeiro quadro de dados, use a assinatura do quadro do cabeçalho.

- HexHash: uma função que primeiro cria um hash SHA-256 de sua entrada e depois usa a função Hex para codificar o hash.
 - non SignatureHeaders: O DateTime cabeçalho codificado como uma string.
 - payload: o buffer de bytes que contém os dados do evento de áudio.
- c. Derive uma chave de assinatura da sua chave de acesso AWS secreta e use-a para assinar o. `stringToSign` Para aumentar o nível de proteção, a chave derivada é específica para a data, o serviço e a Região da AWS. Para obter mais informações, consulte [Calcular a assinatura](#) para o AWS Signature Version 4.

Implemente a função `GetSignatureKey` para derivar sua chave de assinatura. Se você ainda não derivou uma chave de assinatura, consulte [exemplos sobre como derivar uma chave de assinatura para o Signature Version 4](#).

```
String signature = HMACSHA256(derivedSigningKey, stringToSign);
```

- HMACSHA256: uma função que cria uma assinatura usando a função de hash SHA-256.
- derivado `SigningKey`: a chave de assinatura Signature Version 4.
- `string ToSign`: a string que você calculou para o quadro de dados.

Depois de calcular a assinatura do quadro de dados, crie um buffer de bytes que contenha a data, a assinatura e a carga útil do evento de áudio. Envie a matriz de bytes para o Amazon Transcribe para transcrição.

5. Para indicar que o fluxo de áudio está completo, envie um quadro final (um quadro de dados vazio) que contenha apenas a data e a assinatura. Você cria esse quadro final da mesma forma que constrói um quadro de dados.

Amazon Transcribe responde com um fluxo de eventos de transcrição, enviado ao seu aplicativo. Essa resposta é codificada por fluxo de eventos. Ela contém o prelúdio padrão e os seguintes cabeçalhos:

Comprimento em bytes do nome do cabeçalho	Nome do cabeçalho (string)	Tipo de valor de cabeçalho	Comprimento em bytes da string do valor	String do valor (UTF-8)
13	:content-type	7	16	application/json
11	:event-type	7	15	TranscriptEvent
13	:message-type	7	5	evento

Os eventos são enviados no formato de byte bruto. Neste exemplo, os bytes são codificados em base64.

```
AAAAUwAAAEP1RHpYBTpkYXR1CAAAAWiXUkMLEDpjaHVuay1zaWduYXR1cmUGACct6Zy+uymwEK2Srlp/
zVBI
5eGn83jdBwCaRUBJA+eaDafqjqI=
```

Para ver os resultados da transcrição, decodifique os bytes brutos usando a codificação de fluxo de eventos.

```
:content-type: "application/vnd.amazon.eventstream"
:event-type: "TranscriptEvent"
:message-type: "event"

{
  "Transcript":
    {
      "Results":
        [
          results
        ]
    }
}
```

- Para encerrar sua transmissão, envie um evento de áudio vazio para o Amazon Transcribe. Crie o evento de áudio exatamente como qualquer outro, mas com uma carga vazia. Assine o evento e inclua a assinatura no cabeçalho `:chunk-signature`, da seguinte forma:

```
:date: 2019-01-29T01:56:17.291Z
:chunk-signature: signature
```

Tratamento de erros de streaming HTTP/2

Se ocorrer um erro ao processar seu stream de mídia, Amazon Transcribe envia uma resposta de exceção. A resposta tem codificação de fluxo de evento.

A resposta contém o prelúdio padrão e os seguintes cabeçalhos:

Comprimento em bytes do nome do cabeçalho	Nome do cabeçalho (string)	Tipo de valor de cabeçalho	Comprimento em bytes da string do valor	String do valor (UTF-8)
13	:content-type	7	16	application/json
11	:event-type	7	19	BadRequestException
13	:message-type	7	9	exception

Quando a resposta de exceção é decodificada, ela contém as seguintes informações:

```
:content-type: "application/vnd.amazon.eventstream"
:event-type: "BadRequestException"
:message-type: "exception"
```

Exception message

Configurando um WebSocket stream

Os principais componentes de um [WebSocketprotocolo](#) para streaming de solicitações de transcrição Amazon Transcribe são:

- A solicitação de upgrade. Ele contém os parâmetros de consulta para sua solicitação e uma assinatura que é Amazon Transcribe usada como assinatura inicial para assinar os quadros de dados.

- Um ou mais quadros de mensagens na [codificação de fluxo de eventos](#) que contêm metadados e bytes de áudio bruto.
- Um quadro final. Essa é uma mensagem assinada na [codificação de fluxo de eventos](#) com um corpo vazio.

Note

Amazon Transcribe suporta apenas um stream por WebSocket sessão. Se você tentar usar vários fluxos, sua solicitação de transcrição falhará.

1. Anexe a política a seguir à IAM função que faz a solicitação. Consulte [Adicionar IAM políticas](#) para obter mais informações.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "my-transcribe-websocket-policy",
      "Effect": "Allow",
      "Action": "transcribe:StartStreamTranscriptionWebSocket",
      "Resource": "*"
    }
  ]
}
```

2. Para iniciar a sessão, crie um URL pré-assinado no formato a seguir. Quebras de linha foram adicionadas para uma melhor leitura.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=access-key%2FYYYYMMDD%2Fus-west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=YYYYMMDDTHHMMSSZ
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&language-code=en-US
&media-encoding=flac
```



```
&sample-rate=16000
```

Note

O valor máximo para `X-Amz-Expires` é 300 (5 minutos).

Operações e parâmetros adicionais estão listados na [Referência da API](#). Os parâmetros comuns a todas as operações de API da AWS estão listados na seção [Common Parameters](#).

Para construir o URL da solicitação e criar a [assinatura Signature Version 4](#), consulte as etapas a seguir. Os exemplos estão em pseudocódigo.

- a. Cria uma solicitação canônica. Uma solicitação canônica é uma string que inclui informações da solicitação em um formato padronizado. Isso garante que, ao AWS receber a solicitação, ela possa calcular a mesma assinatura que você criou para sua URL. Para obter mais informações, consulte [Criar uma solicitação canônica](#) para o Signature Version 4.

```
# HTTP verb
method = "GET"
# Service name
service = "transcribe"
# Region
region = "us-west-2"
# Amazon Transcribe streaming endpoint
endpoint = "wss://transcribestreaming.us-west-2.amazonaws.com:8443"
# Host
host = "transcribestreaming.us-west-2.amazonaws.com:8443"
# Date and time of request
amz-date = YYYYMMDDTHHMMSSZ
# Date without time for credential scope
datestamp = YYYYMMDD
```

- b. Crie um URI canônico, que é a parte do URI entre o domínio e a string de consulta.

```
canonical_uri = "/stream-transcription-websocket"
```

- c. Crie cabeçalhos canônicos e cabeçalhos assinados. Observe o `\n` que aparece no final dos cabeçalhos canônicos.
 - Acrescente o nome do cabeçalho em letras minúsculas seguido de dois-pontos (:).

- Anexe uma lista separada por vírgulas de valores para esse cabeçalho. Não classifique valores em cabeçalhos que tenham vários valores.
- Anexe uma nova linha (`\n`).

```
canonical_headers = "host:" + host + "\n"  
signed_headers = "host"
```

- d. Faça uma correspondência entre o algoritmo e o algoritmo de hash. Usar SHA-256.

```
algorithm = "AWS4-HMAC-SHA256"
```

- e. Crie o escopo da credencial, que define o escopo da chave derivada para a data, a Região da AWS e o serviço. Por exemplo, `20220127/us-west-2/transcribe/aws4_request`.

```
credential_scope = datestamp + "/" + region + "/" + service + "/" +  
"aws4_request"
```

- f. Crie a string de consulta canônica. Os valores de string de consulta devem ser codificados por URI e classificados por nome.
- Classifique os nomes de parâmetro por ponto de código de caracteres em ordem ascendente. Parâmetros com nomes duplicados devem ser classificados pelo valor. Por exemplo, um nome de parâmetro que começa com a letra maiúscula F precede um nome de parâmetro que começa com a letra minúscula b.
 - Não codificar em URI nenhum dos caracteres não reservados definidos pela RFC 3986: A-Z, a-z, 0-9, hífen (-), sublinhado (_), ponto (.) e til (~).
 - Codifique em percentual todos os outros caracteres com `%XY`, em que X e Y são caracteres hexadecimais (de 0 a 9 e maiúsculas de A a F). Por exemplo, o caractere de espaço deve ser codificado como `%20` (não inclua "+", como fazem alguns esquemas de codificação); os caracteres UTF-8 estendidos devem estar no formato `%XY%ZA%BC`.
 - Codifique duas vezes todos os caracteres de sinal de igual (=) em valores de parâmetro.

```
canonical_querystring = "X-Amz-Algorithm=" + algorithm  
canonical_querystring += "&X-Amz-Credential=" + URI-encode(access key + "/" +  
credential_scope)  
canonical_querystring += "&X-Amz-Date=" + amz_date  
canonical_querystring += "&X-Amz-Expires=300"
```

```
canonical_querystring += "&X-Amz-Security-Token=" + token
canonical_querystring += "&X-Amz-SignedHeaders=" + signed_headers
canonical_querystring += "&language-code=en-US&media-encoding=flac&sample-
rate=16000"
```

- g. Crie um hash da carga útil. Para uma solicitação GET, a carga útil é uma string vazia.

```
payload_hash = HashSHA256("").Encode("utf-8").HexDigest()
```

- h. Combine os elementos a seguir para criar a solicitação canônica.

```
canonical_request = method + '\n'
+ canonical_uri + '\n'
+ canonical_querystring + '\n'
+ canonical_headers + '\n'
+ signed_headers + '\n'
+ payload_hash
```

3. Crie a string para assinar, que contém metainformações sobre a solicitação. Você usará a string para assinar na próxima etapa, quando calcular a assinatura da solicitação. Para obter mais informações, consulte [Criar uma string para assinar](#) para o Signature Version 4.

```
string_to_sign=algorithm + "\n"
+ amz_date + "\n"
+ credential_scope + "\n"
+ HashSHA256(canonical_request.Encode("utf-8")).HexDigest()
```

4. Calcule a assinatura. Para fazer isso, obtenha uma chave de assinatura da sua chave de acesso AWS secreta. Para aumentar o nível de proteção, a chave derivada é específica para a data, o serviço e a Região da AWS. Use essa chave derivada para assinar a solicitação. Para obter mais informações, consulte [Calcular a assinatura para a versão 4 do AWS Signature](#).

Implemente a função `GetSignatureKey` para derivar sua chave de assinatura. Se você ainda não derivou uma chave de assinatura, consulte [exemplos sobre como derivar uma chave de assinatura para o Signature Version 4](#).

```
#Create the signing key
signing_key = GetSignatureKey(secret_key, timestamp, region, service)

# Sign the string_to_sign using the signing key
```

```
signature = HMAC.new(signing_key, (string_to_sign).Encode("utf-8"),  
    Sha256()).HexDigest
```

A função `HMAC(key, data)` representa uma função HMAC-SHA256 que retorna resultados em formato binário.

5. Adicione informações de assinatura à solicitação e crie o URL da solicitação.

Depois de calcular a assinatura, adicione-a à string de consulta. Para obter mais informações, consulte [Adicionar a assinatura à solicitação](#).

Primeiro, adicione as informações de autenticação à string de consulta.

```
canonical_querystring += "&X-Amz-Signature=" + signature
```

Segundo, crie o URL para a solicitação.

```
request_url = endpoint + canonical_uri + "?" + canonical_querystring
```

Use o URL da solicitação com sua WebSocket biblioteca para fazer a solicitação para Amazon Transcribe.

6. A solicitação Amazon Transcribe deve incluir os seguintes cabeçalhos. Normalmente, esses cabeçalhos são gerenciados pela sua biblioteca WebSocket cliente.

```
Host: transcribestreaming.us-west-2.amazonaws.com:8443  
Connection: Upgrade  
Upgrade: websocket  
Origin: URI-of-WebSocket-client  
Sec-WebSocket-Version: 13  
Sec-WebSocket-Key: randomly-generated-string
```

7. Quando Amazon Transcribe recebe sua WebSocket solicitação, ela responde com uma resposta de WebSocket upgrade. Normalmente, sua WebSocket biblioteca gerencia essa resposta e configura um soquete para comunicação com Amazon Transcribe.

A seguir está a resposta de Amazon Transcribe. Quebras de linha foram adicionadas para uma melhor leitura.

```
HTTP/1.1 101 WebSocket Protocol Handshake
```

```

Connection: upgrade
Upgrade: websocket
websocket-origin: wss://transcribestreaming.us-west-2.amazonaws.com:8443
websocket-location: transcribestreaming.us-west-2.amazonaws.com:8443/stream-
transcription-websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-west-2%2Ftranscribe
%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&language-code=en-US
&session-id=String
&media-encoding=flac
&sample-rate=16000
x-amzn-RequestId: RequestId
Strict-Transport-Security: max-age=31536000
sec-websocket-accept: hash-of-the-Sec-WebSocket-Key-header

```

8. Faça sua solicitação WebSocket de streaming.

Depois que a WebSocket conexão é estabelecida, o cliente pode começar a enviar uma sequência de quadros de áudio, cada um codificado usando a codificação de [fluxo de eventos](#).

Cada quadro de dados contém três cabeçalhos combinados com um bloco de bytes de áudio bruto. A tabela a seguir descreve esses cabeçalhos.

Comprimento em bytes do nome do cabeçalho	Nome do cabeçalho (string)	Tipo de valor de cabeçalho	Comprimento em bytes da string do valor	String do valor (UTF-8)
13	:content-type	7	24	application/octet-stream
11	:event-type	7	10	AudioEvent
13	: message-type	7	5	evento

9. Para encerrar o fluxo de dados, envie um bloco de áudio vazio em uma mensagem codificada de fluxo de eventos.

A resposta contém bytes brutos com codificação de fluxo de evento na carga. Ela contém o prelúdio padrão e os seguintes cabeçalhos:

Comprimento em bytes do nome do cabeçalho	Nome do cabeçalho (string)	Tipo de valor de cabeçalho	Comprimento em bytes da string do valor	String do valor (UTF-8)
13	:content-type	7	16	application/json
11	:event-type	7	15	TranscriptEvent
13	: message-type	7	5	evento

Ao decodificar a resposta binária, você obtém uma estrutura JSON que contém os resultados da transcrição.

Tratamento de erros WebSocket de streaming

Se ocorrer uma exceção ao processar sua solicitação, Amazon Transcribe responderá com um WebSocket quadro de terminal contendo uma resposta codificada de fluxo de eventos. Essa resposta contém os cabeçalhos descritos na tabela a seguir; o corpo da resposta contém uma mensagem de erro descritiva. Depois de enviar a resposta da exceção, Amazon Transcribe envia um quadro fechado.

Comprimento em bytes do nome do cabeçalho	Nome do cabeçalho (string)	Tipo de valor de cabeçalho	Comprimento em bytes da string do valor	String do valor (UTF-8)
13	:content-type	7	16	application/json
15	:exception-type	7	varia	varia, veja abaixo
13	: message-type	7	9	exception

O cabeçalho `exception-type` contém um dos seguintes valores:

- **BadRequestException:** houve um erro do cliente quando o fluxo foi criado ou ocorreu um erro durante o fluxo de dados. Certifique-se de que seu cliente esteja pronto para aceitar dados e tente sua solicitação novamente.
- **InternalFailureException:** Amazon Transcribe teve um problema durante o aperto de mão com o cliente. Tente sua solicitação novamente.
- **LimitExceededException:** o cliente excedeu o limite de fluxo simultâneo. Para obter mais informações, consulte [Amazon Transcribe Limites](#). Reduza o número de fluxos que você está transcrevendo.
- **UnrecognizedClientException:** a solicitação de WebSocket upgrade foi assinada com uma chave de acesso ou chave secreta incorreta. Verifique se você está criando corretamente a chave de acesso e tente fazer a solicitação novamente.

Amazon Transcribe também pode retornar qualquer um dos erros comuns de serviço. Para obter uma lista, consulte [Erros comuns](#).

Codificação de fluxo de eventos

Amazon Transcribe usa um formato chamado codificação de fluxo de eventos para transcrições de streaming.

A codificação de fluxo de eventos fornece comunicação bidirecional entre um cliente e um servidor. Os quadros de dados enviados para o serviço de Amazon Transcribe streaming são codificados nesse formato. A resposta de Amazon Transcribe também usa essa codificação.

Cada mensagem consiste em duas seções: o prelúdio e os dados. O prelúdio consiste em:

1. O comprimento de byte total da mensagem
2. O comprimento total em bytes de todos os cabeçalhos

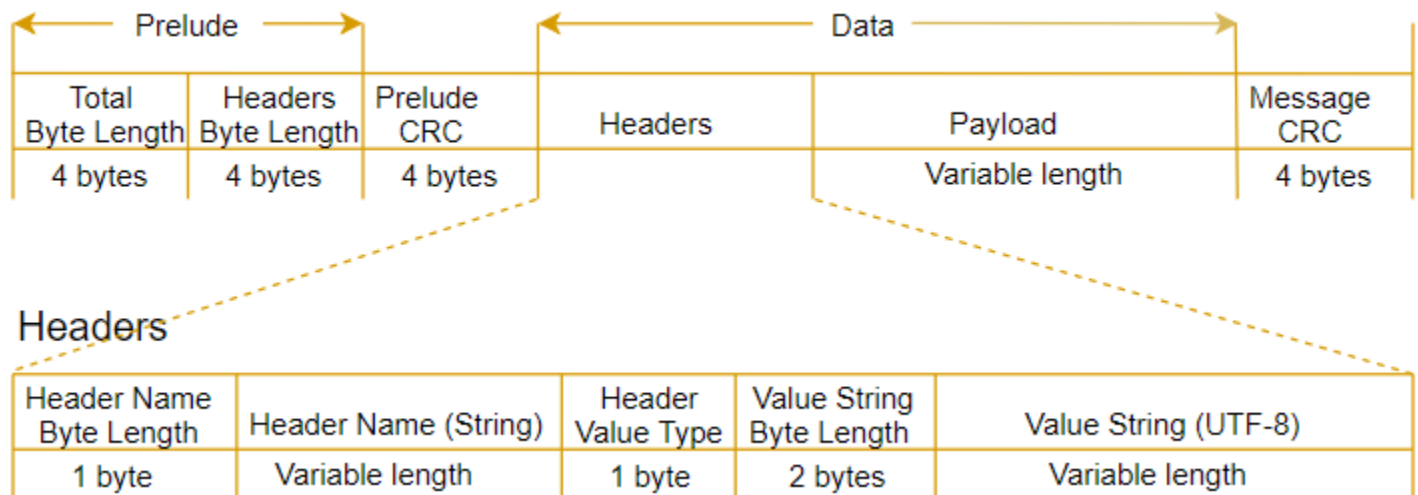
A seção de dados consiste em:

1. Cabeçalhos
2. Carga útil

Cada seção termina com uma soma de verificação de redundância cíclica (CRC) inteira de 4 bytes big-endian. A soma de verificação CRC da mensagem é para a seção de prelúdio e para a seção de dados. O Amazon Transcribe usa CRC32 (muitas vezes chamado de GZIP CRC32) para calcular ambas as CRCs. Para mais informações sobre CRC32, consulte [Especificação do formato de arquivo GZIP versão 4.3](#).

O total de sobrecarga de mensagem, incluindo o prelúdio e ambas as somas de verificação, é de 16 bytes.

O diagrama a seguir mostra os componentes que formam uma mensagem e um cabeçalho. Há vários cabeçalhos por mensagem.



Cada mensagem contém os seguintes componentes:

- **Prelúdio:** consiste em dois campos de 4 bytes, para um total fixo de 8 bytes.
 - Primeiros 4 bytes: o comprimento em bytes do inteiro big-endian da mensagem inteira, incluindo esse campo de comprimento de 4 bytes.
 - Segundo 4 bytes: o comprimento em bytes do inteiro big-endian da parte “headers” da mensagem, excluindo o próprio campo de comprimento “headers”.
- **CRC do prelúdio:** a soma de verificação CRC de 4 bytes para a porção do prelúdio da mensagem, excluindo a própria CRC. O prelúdio tem um CRC separado da mensagem CRC. Isso garante que seja possível detectar informações corrompidas de comprimento de bytes imediatamente sem causar erros, como sobrecarga de buffer.
- **Cabeçalhos:** metadados que anotam a mensagem; por exemplo, tipo de mensagem e tipo de conteúdo. As mensagens têm vários cabeçalhos, os quais são pares de chave-valor em que a

chave é uma string UTF-8. Os cabeçalhos podem aparecer em qualquer ordem na parte “headers” da mensagem, e cada cabeçalho pode aparecer apenas uma vez.

- Carga útil: o conteúdo de áudio a ser transcrito.
- CRC da mensagem: a soma de verificação CRC de 4 bytes desde o início da mensagem até o início da soma de verificação. Ou seja, tudo na mensagem, exceto a CRC em si.

O quadro do cabeçalho é o quadro de autorização para a transcrição do streaming. Amazon Transcribe usa o valor do cabeçalho de autorização como semente para gerar uma cadeia de cabeçalhos de autorização para os quadros de dados na solicitação.

Cada cabeçalho contém os componentes a seguir; há vários cabeçalhos por quadro.

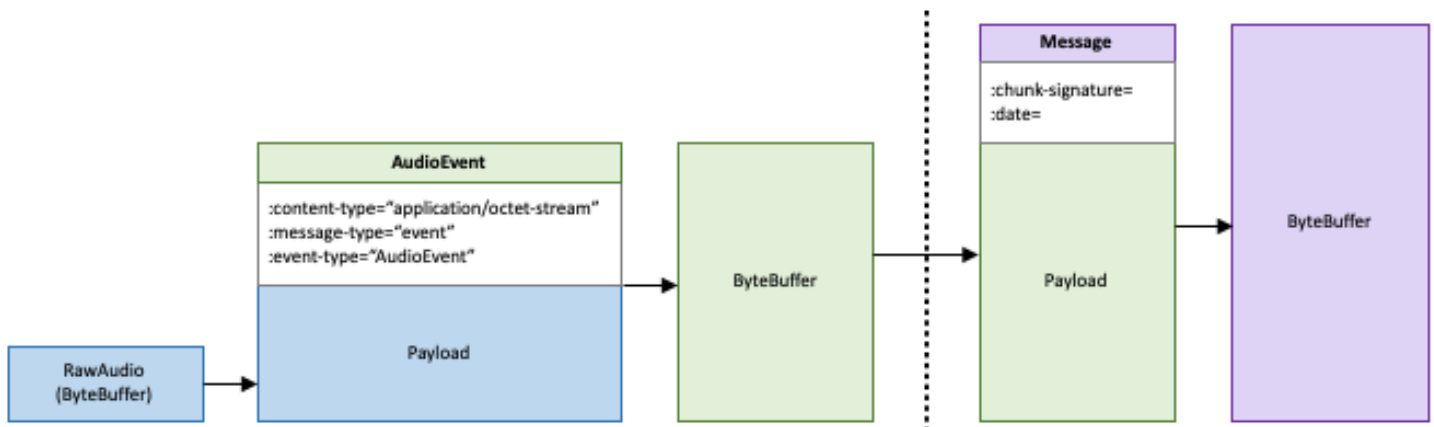
- O comprimento de byte do nome do cabeçalho: o comprimento de bytes do nome do cabeçalho.
- Nome do cabeçalho: o nome do cabeçalho que indica o tipo de cabeçalho. Para valores válidos, consulte as seguintes descrições de quadro.
- Tipo de valor do cabeçalho: um número que indica o valor do cabeçalho. A lista a seguir mostra os valores possíveis para o cabeçalho e o que eles indicam.
 - 0: TRUE
 - 1: FALSE
 - 2: BYTE
 - 3: SHORT
 - 4: INTEGER
 - 5: LONG
 - 6: BYTE ARRAY
 - 7: STRING
 - 8: TIMESTAMP
 - 9: UUID
- Comprimento em bytes da string: o comprimento em bytes da string do cabeçalho.
- Valor do cabeçalho: o valor da string do cabeçalho. Os valores válidos para esse campo dependem do tipo de cabeçalho. Consulte [Configurar um fluxo de HTTP/2](#) ou [Configurando um WebSocket stream](#) para obter mais informações.

Quadros de dados

Cada solicitação de streaming contém um ou mais quadros de dados. Existem duas etapas para criar um quadro de dados:

1. Combine dados de áudio brutos com metadados para criar a carga útil da solicitação.
2. Combine a carga com uma assinatura para formar a mensagem de evento enviada para o Amazon Transcribe.

O diagrama a seguir mostra como isso funciona.



Fila de Job

Usando a fila de trabalhos, você pode enviar mais solicitações de transcrição do que as que podem ser processadas simultaneamente. Sem a fila de trabalhos, depois de atingir a cota de solicitações simultâneas permitidas, você deve esperar até que uma ou mais solicitações sejam concluídas antes de enviar uma nova solicitação.

A fila de Job é opcional para solicitações de trabalho de transcrição. As solicitações de análise pós-chamada têm o enfileiramento de tarefas ativado automaticamente.

Se você habilitar o enfileiramento de trabalhos, Amazon Transcribe criará uma fila que contém todas as solicitações que excedem seu limite. Assim que uma solicitação é concluída, uma nova solicitação é retirada da sua fila e processada. As solicitações em fila são processadas em uma ordem FIFO (primeiro a entrar, primeiro a sair).

Você pode adicionar até 10.000 trabalhos à sua fila. Se você exceder esse limite, receberá um `LimitExceededConcurrentJobException` erro. Para manter o desempenho ideal, use Amazon Transcribe apenas até 90% da sua cota (uma taxa de largura de banda de 0,9) para processar trabalhos em fila. Observe que esses são valores padrão que podem ser aumentadas mediante solicitação.

Tip

Você pode encontrar uma lista de limites e cotas padrão para Amazon Transcribe recursos na [Referência AWS Geral](#). Alguns desses padrões podem ser aumentadas mediante solicitação.

Se você habilitar o enfileiramento de trabalhos, mas não exceder a cota para solicitações simultâneas, todas as solicitações serão processadas simultaneamente.

Habilitação de trabalhos em lote

Você pode habilitar o enfileiramento de trabalhos usando os AWSSDKs AWS Management Console AWS CLI, ou os SDKs; veja exemplos a seguir; veja os exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).

2. No painel de navegação, escolha Trabalhos de transcrição e selecione Criar tarefa (canto superior direito). Isso abre a página Especificar detalhes do trabalho.
3. Na caixa Configurações do Job, há um painel de configurações adicionais. Se você expandir esse painel, poderá selecionar a caixa Adicionar à fila de trabalhos para ativar o enfileiramento de trabalhos.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)

If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)

If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

▼ Additional settings

Job queue - optional [Info](#)

Enables you to submit jobs beyond the limit for concurrent jobs (100). You must specify access permissions to the resources that job queuing uses.

Add to job queue

4. Preencha os outros campos que você deseja incluir na página Especificar detalhes do trabalho e selecione Avançar. Isso leva você à página Configurar tarefa - opcional.

5. Selecione Criar tarefa para executar sua tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o `job-execution-settings` parâmetro com o `AllowDeferredExecution` subparâmetro. Observe que, ao incluir `AllowDeferredExecution` em sua solicitação, você também deve incluir `DataAccessRoleArn`.

Para obter mais informações, consulte [StartTranscriptionJob](#) e [JobExecutionSettings](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--job-execution-settings  
  AllowDeferredExecution=true,DataAccessRoleArn=arn:aws:iam::111122223333:role/  
ExampleRole
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que permite o enfileiramento.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-queueing-request.json
```

O arquivo `my-first-queueing-request.json` contém o seguinte corpo da solicitação.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "JobExecutionSettings": {  
    "AllowDeferredExecution": true,  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole"  
  }  
}
```

```
}  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para habilitar o enfileiramento de trabalhos usando o `AllowDeferredExecution` argumento para o método [start_transcription_job](#). Observe que, ao incluir `AllowDeferredExecution` em sua solicitação, você também deve incluir `DataAccessRoleArn`. Para obter mais informações, consulte [StartTranscriptionJob](#) e [JobExecutionSettings](#).

Para obter exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e serviços cruzados, consulte o [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#) capítulo.

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-queueing-request"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri  
    },  
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',  
    OutputKey = 'my-output-files/',  
    LanguageCode = 'en-US',  
    JobExecutionSettings = {  
        'AllowDeferredExecution': True,  
        'DataAccessRoleArn': 'arn:aws:iam::111122223333:role/ExampleRole'  
    }  
)  
  
while True:  
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)  
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:  
        break  
    print("Not ready yet...")  
    time.sleep(5)  
print(status)
```

Você pode ver o progresso de um trabalho em fila por meio do AWS Management Console ou enviando uma [GetTranscriptionJobs](#) solicitação. Quando um trabalho está na fila, o status é QUEUED. O status muda para IN_PROGRESS quando seu trabalho começa a ser processado e, em seguida, muda para COMPLETED ou FAILED quando o processamento é concluído.

Marcar recursos

Uma tag é um rótulo de metadados personalizado que você pode adicionar a um recurso para facilitar a identificação, a organização e a localização em uma pesquisa. As tags são compostas por duas partes individuais: uma chave de tag e um valor de tag. Isso é chamado de par chave:valor.

Uma chave de tag normalmente representa uma categoria maior, enquanto um valor de tag representa um subconjunto dessa categoria. Por exemplo, você poderia ter a tag `key=color` e a tag `value=blue`, o que produziria o par chave:valor `Color:Blue`. Observe que você pode definir o valor de uma tag a uma string vazia, mas não pode configurar o valor de um tag como nula. Omitir o valor da tag é o mesmo que usar uma string vazia.

Tip

AWS Billing and Cost Management pode usar etiquetas para separar suas contas em categorias dinâmicas. Por exemplo, se você adicionar tags para representar diferentes departamentos da sua empresa, como `Department:Sales` ou `Department:Legal`, AWS poderá fornecer sua distribuição de custos por departamento.

No Amazon Transcribe, você pode marcar os seguintes recursos:

- Trabalhos de transcrição
- Empregados de transcrição
- Trabalhos de transcrição pós-chamada de análise de chamadas
- Vocabulário personalizado de vocabulário personalizado
- Vocabulários médicos personalizados
- Filtros de vocabulário personalizados
- Modelos de linguagem personalizados

As chaves de tag podem ter até 128 caracteres e os valores de tag podem ter até 256 caracteres; ambas diferenciam maiúsculas e minúsculas. Amazon Transcribe suporta até 50 etiquetas por recurso. Em um recurso específico, cada chave de tag deve ser exclusiva com apenas um valor. Observe que suas tags não podem começar com `aws:` porque AWS reserva esse prefixo para tags geradas pelo sistema. Você não pode adicionar, modificar ou excluir `aws:*` tags, e elas não contam para seu `tags-per-resource` limite.

Operações de API específicas para marcação de recursos

[ListTagsForResource](#), [TagResource](#), [UntagResource](#)

Para usar as APIs de marcação, inclua um nome de recurso da Amazon Resource (ARN) na solicitação do. Os ARNs têm o formato `arn:partition:service:region:account-id:resource-type/resource-id`. Por exemplo, o ARN associado a um trabalho de transcrição pode ser parecido com `arn:aws:transcribe:us-west-2:111122223333:transcription-job/my-transcription-job-name`.

Para saber mais sobre marcação, incluindo as melhores práticas, consulte [AWS Recursos de marcação](#).

Controle de acesso com base em tags

Você pode usar etiquetas para controlar o acesso ao Contas da AWS. Para controle de acesso baseado em tags, forneça informações sobre a tag no elemento de condição de uma IAM política. Em seguida, use as tags e a chave de condição da tag associada para controlar o acesso a:

- Recursos: controle o acesso aos Amazon Transcribe recursos de acordo com as tags atribuídas a esses recursos do.
 - Use a chave `aws:ResourceTag/key-name` condição para especificar qual par de chave-valor da tag deve ser anexado ao recurso.
- Solicitação: controle quais tags podem ser transmitidas em uma solicitação.
 - Use a chave `aws:RequestTag/key-name` condição para especificar quais tags podem ser adicionadas, modificadas ou removidas de um IAM usuário ou de uma função do.
- Processos de autorização: controle o acesso baseado em tags para qualquer parte do seu processo de autorização.
 - Use a chave `aws:TagKeys/` condição para controlar quais chaves de etiqueta específicas podem ser usadas em um recurso, em uma solicitação ou por uma entidade de segurança. Neste caso, o valor da chave não importa.

Para obter um exemplo de política de controle de acesso baseada em tags, consulte [Visualização de trabalhos de transcrição com base em tags](#).

Para obter informações mais detalhadas sobre o controle de acesso baseado em tags, consulte [Controlar o acesso aos AWS recursos usando tags](#).

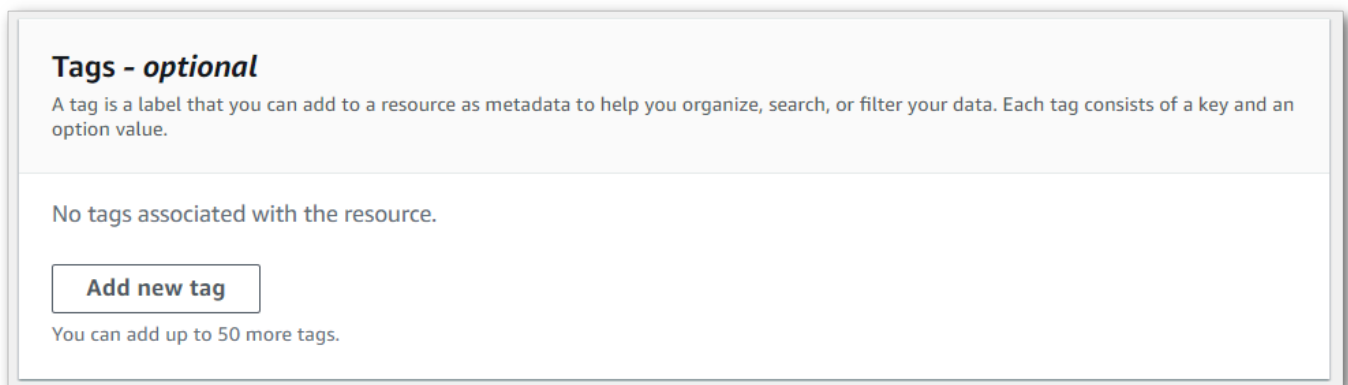
Adicionar etiquetas aos Amazon Transcribe recursos do

Você pode adicionar tags antes ou depois de executar seu Amazon Transcribe trabalho. Usando as APIs Create* e Start* existentes, você pode adicionar tags à sua solicitação de transcrição.

Você pode adicionar, modificar ou excluir tags usando os AWS Management Console AWS CLI, ou AWSSDKs; veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Trabalhos de transcrição e selecione Criar tarefa (canto superior direito). Isso abre a página Especificar detalhes do trabalho.
3. Role até a parte inferior da página Especificar detalhes do trabalho para encontrar a caixa Tags - opcional e selecione Adicionar nova tag.



4. Insira informações para o campo Chave e, opcionalmente, para o campo Valor.

Tags - optional

A tag is a label that you can add to a resource as metadata to help you organize, search, or filter your data. Each tag consists of a key and an option value.

Key

Value - optional

You can add up to 49 more tags.

- Preencha os outros campos que você deseja incluir na página Especificar detalhes do trabalho e selecione Avançar. Isso leva você à página Configurar tarefa - opcional.

Selecione Criar tarefa para executar sua tarefa de transcrição.

- Você pode visualizar as tags associadas a um trabalho de transcrição navegando até a página Trabalhos de transcrição, selecionando um trabalho de transcrição e rolando até a parte inferior da página de informações desse trabalho. Se você quiser editar as tags, poderá fazê-lo selecionando Gerenciar tags.

Tags (2)		<input type="button" value="Manage Tags"/>
Key		Value
color		blue

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o `Tags` parâmetro. Para obter mais informações, consulte [StartTranscriptionJob](#) e [Tag](#).

```
aws transcribe start-transcription-job \
--region us-west-2 \
--transcription-job-name my-first-transcription-job \
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \
--output-bucket-name DOC-EXAMPLE-BUCKET \
--output-key my-output-files/ \
--language-code en-US \
```

```
--tags Key=color,Value=blue Key=shape,Value=square
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que adiciona tags a essa tarefa.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-first-tagging-job.json
```

O arquivo `my-first-tagging-job.json` contém o seguinte corpo da solicitação.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Tags": [  
    {  
      "Key": "color",  
      "Value": "blue"  
    },  
    {  
      "Key": "shape",  
      "Value": "square"  
    }  
  ]  
}
```

AWS SDK for Python (Boto3)

O exemplo a seguir usa o AWS SDK for Python (Boto3) para adicionar uma tag usando o `Tags` argumento para o método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#) e [Tag](#).

Para obter exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e serviços cruzados, consulte o [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#) capítulo.

```
from __future__ import print_function
```

```
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Tags = [
        {
            'Key': 'color',
            'Value': 'blue'
        }
    ]
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Separação de locutores (diarização)

Com a diarização de alto-falantes, você pode distinguir entre diferentes alto-falantes em sua saída de transcrição. Amazon Transcribe pode diferenciar entre no máximo 30 alto-falantes exclusivos e rotula o texto de cada locutor exclusivo com um valor exclusivo (spk_0 por meio de spk_9).

Além das [seções de transcrição padrão](#) (transcripts e items), as solicitações com a separação de locutores habilitada incluem uma seção speaker_labels. Esta seção é agrupada por locutor e contém informações sobre cada enunciado, incluindo o rótulo do locutor e os respectivos registros de data e hora.

```
"speaker_labels": {
  "channel_label": "ch_0",
  "speakers": 2,
  "segments": [
    {
      "start_time": "4.87",
      "speaker_label": "spk_0",
      "end_time": "6.88",
      "items": [
        {
          "start_time": "4.87",
          "speaker_label": "spk_0",
          "end_time": "5.02"
        },
        ...
      ]
    },
    {
      "start_time": "8.49",
      "speaker_label": "spk_1",
      "end_time": "9.24",
      "items": [
        {
          "start_time": "8.49",
          "speaker_label": "spk_1",
          "end_time": "8.88"
        },
        ...
      ]
    },
    ...
  ]
}
```

Para ver um exemplo completo de transcrição com separação de locutores (para dois locutores), consulte [Exemplo de saída de diarização \(lote\)](#).

Separar locutores em uma transcrição em lote

Para separar locutores em uma transcrição em lote, veja os seguintes exemplos:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Tarefas de transcrição e selecione Criar tarefa (no canto superior direito). Isso abre a página Especificar os detalhes da tarefa.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

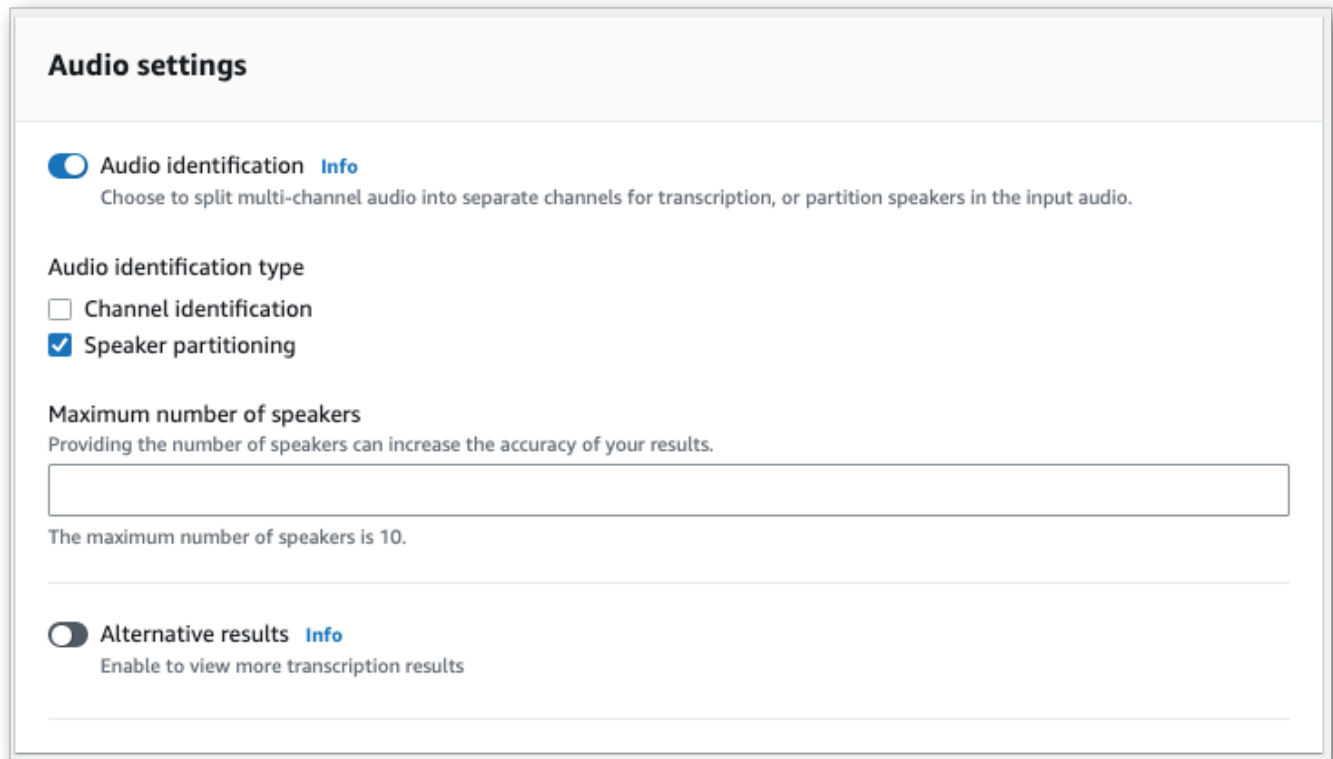
Language

Choose the language of the input audio.

► **Additional settings**

3. Preencha todos os campos que você deseja incluir na página Especificar os detalhes da tarefa e selecione Próximo. Isso leva você à página Configurar tarefa - opcional.

No painel de Configurações de áudio, selecione Separação de oradores (sob o título “Tipo de identificação de áudio”). Como opção, você pode especificar o número de locutores (no máximo, dez) que deseja separar.



Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or partition speakers in the input audio.

Audio identification type

Channel identification

Speaker partitioning

Maximum number of speakers
Providing the number of speakers can increase the accuracy of your results.

The maximum number of speakers is 10.

Alternative results [Info](#)
Enable to view more transcription results

4. Selecione Criar tarefa para executar a tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#). Para ter mais informações, consulte [StartTranscriptionJob](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--show-speaker-labels TRUE \  

```



```
--max-speaker-labels 3
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que permite o particionamento de alto-falantes com esse trabalho.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-transcription-job.json
```

O arquivo `my-first-transcription-job.json` contém o seguinte corpo da solicitação.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "ShowSpeakerLabels": 'TRUE',  
  "MaxSpeakerLabels": 3  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para identificar canais usando o método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#).

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-transcription-job"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri  
    },  
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',  
    OutputKey = 'my-output-files/',
```

```
LanguageCode = 'en-US',
Settings = {
    'ShowSpeakerLabels': True,
    'MaxSpeakerLabels': 3
}
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Separar locutores em uma transcrição de streaming

Para separar locutores em uma transcrição de streaming, veja os seguintes exemplos:

Transcrições de streaming

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Transcrição em tempo real. Role para baixo até Configurações de áudio e expanda esse campo se ele estiver minimizado.

Real-time transcription [Info](#)

See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.

Transcription

[Download full transcript](#)[Start streaming](#)

Transcription output

Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone

00:00 of 15:00 min audio stream

▶ **Language settings**

▼ **Audio settings**

Speaker partitioning [Info](#)

Partition the different speakers in the stream. Speaker partitioning might vary in availability between languages.

▶ **Content removal settings**

▶ **Customizations**

3. Ative Separação de oradores.

▶ **Language settings**

▼ **Audio settings**

Speaker partitioning [Info](#)

Partition the different speakers in the stream. Speaker partitioning might vary in availability between languages.

▶ **Content removal settings**

▶ **Customizations**

4. Agora está tudo pronto para transcrever seu fluxo. Selecione Iniciar transmissão e comece a falar. Para finalizar o ditado, selecione Interromper transmissão.

Fluxo HTTP/2

Este exemplo cria uma solicitação HTTP/2 que separa os locutores na saída da transcrição. Para obter mais informações sobre como usar o streaming HTTP/2 com Amazon Transcribe, consulte [Configurar um fluxo de HTTP/2](#). Para obter mais detalhes sobre parâmetros e cabeçalhos específicos de Amazon Transcribe, consulte [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-show-speaker-label: true
transfer-encoding: chunked
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

WebSocket stream

Este exemplo cria um URL pré-assinado que separa os locutores na saída da transcrição. Quebras de linha foram adicionadas para uma melhor leitura. Para obter mais informações sobre o uso de WebSocket streams com Amazon Transcribe, consulte [Configurando um WebSocket stream](#). Para obter mais detalhes sobre os parâmetros, consulte [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
```

```
&language-code=en-US
&specialty=PRIMARYCARE
&type=DICTION
&media-encoding=flac
&sample-rate=16000
&show-speaker-label=true
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Exemplo de saída de diarização (lote)

Aqui está um exemplo de saída para uma transcrição em lote com a diarização ativada.

```
{
  "jobName": "my-first-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "I've been on hold for an hour. Sorry about that."
      }
    ],
    "speaker_labels": {
      "channel_label": "ch_0",
      "speakers": 2,
      "segments": [
        {
          "start_time": "4.87",
          "speaker_label": "spk_0",
          "end_time": "6.88",
          "items": [
            {
              "start_time": "4.87",
              "speaker_label": "spk_0",
              "end_time": "5.02"
            },
            {
              "start_time": "5.02",
              "speaker_label": "spk_0",
              "end_time": "5.17"
            }
          ]
        }
      ]
    }
  }
}
```

```
        "start_time": "5.17",
        "speaker_label": "spk_0",
        "end_time": "5.29"
    },
    {
        "start_time": "5.29",
        "speaker_label": "spk_0",
        "end_time": "5.64"
    },
    {
        "start_time": "5.64",
        "speaker_label": "spk_0",
        "end_time": "5.84"
    },
    {
        "start_time": "6.11",
        "speaker_label": "spk_0",
        "end_time": "6.26"
    },
    {
        "start_time": "6.26",
        "speaker_label": "spk_0",
        "end_time": "6.88"
    }
]
},
{
    "start_time": "8.49",
    "speaker_label": "spk_1",
    "end_time": "9.24",
    "items": [
        {
            "start_time": "8.49",
            "speaker_label": "spk_1",
            "end_time": "8.88"
        },
        {
            "start_time": "8.88",
            "speaker_label": "spk_1",
            "end_time": "9.05"
        },
        {
            "start_time": "9.05",
            "speaker_label": "spk_1",
```

```
                "end_time": "9.24"
            }
        ]
    }
},
"items": [
    {
        "start_time": "4.87",
        "speaker_label": "spk_0",
        "end_time": "5.02",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "I've"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.02",
        "speaker_label": "spk_0",
        "end_time": "5.17",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "been"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.17",
        "speaker_label": "spk_0",
        "end_time": "5.29",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "on"
            }
        ],
        "type": "pronunciation"
    },
    {
```

```
    "start_time": "5.29",
    "speaker_label": "spk_0",
    "end_time": "5.64",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "hold"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.64",
    "speaker_label": "spk_0",
    "end_time": "5.84",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "for"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "6.11",
    "speaker_label": "spk_0",
    "end_time": "6.26",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "an"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "6.26",
    "speaker_label": "spk_0",
    "end_time": "6.88",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "hour"
      }
    ]
  }
```



```
    ],
    "type": "pronunciation"
  },
  {
    "speaker_label": "spk_0",
    "alternatives": [
      {
        "confidence": "0.0",
        "content": "."
      }
    ],
    "type": "punctuation"
  },
  {
    "start_time": "8.49",
    "speaker_label": "spk_1",
    "end_time": "8.88",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "Sorry"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "8.88",
    "speaker_label": "spk_1",
    "end_time": "9.05",
    "alternatives": [
      {
        "confidence": "0.902",
        "content": "about"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "9.05",
    "speaker_label": "spk_1",
    "end_time": "9.24",
    "alternatives": [
      {
        "confidence": "1.0",
```

```
        "content": "that"
      }
    ],
    "type": "pronunciation"
  },
  {
    "speaker_label": "spk_1",
    "alternatives": [
      {
        "confidence": "0.0",
        "content": "."
      }
    ],
    "type": "punctuation"
  }
]
},
"status": "COMPLETED"
}
```

Transcrevendo áudio multicanal

Se o áudio tiver dois canais, você poderá usar a identificação do canal para transcrever a fala de cada canal separadamente. Amazon Transcribe atualmente não suporta áudio com mais de dois canais.

Em sua transcrição, os canais recebem os rótulos `ch_0` e `ch_1`.

Além das [seções de transcrição padrão](#) (`transcriptseitems`), as solicitações com identificação de canal ativada incluem uma `channel_labels` seção. Esta seção contém cada expressão ou sinal de pontuação, agrupado por canal, e seu rótulo de canal associado, carimbos de data e hora e pontuação de confiança.

```
"channel_labels": {
  "channels": [
    {
      "channel_label": "ch_0",
      "items": [
        {
          "channel_label": "ch_0",
          "start_time": "4.86",
          "end_time": "5.01",
          "alternatives": [
            {
              "confidence": "1.0",
              "content": "I've"
            }
          ],
          "type": "pronunciation"
        },
        ...
      ],
      "channel_label": "ch_1",
      "items": [
        {
          "channel_label": "ch_1",
          "start_time": "8.5",
          "end_time": "8.89",
          "alternatives": [
            {
              "confidence": "1.0",
              "content": "Sorry"
            }
          ]
        }
      ]
    }
  ]
}
```

```
        }
      ],
      "type": "pronunciation"
    },
    ...
    "number_of_channels": 2
  },
```

Observe que, se uma pessoa em um canal fala ao mesmo tempo que uma pessoa em um canal separado, os registros de data e hora de cada canal se sobrepõem enquanto os indivíduos estão falando uns sobre os outros.

Para ver um exemplo completo de transcrição com identificação de canal, consulte [Exemplo de saída de identificação de canal \(lote\)](#).

Usando a identificação de canais em uma transcrição em lote

Para identificar canais em uma transcrição em lote, você pode usar os SDKs AWS Management Console, AWS CLI, ou AWS SDKs; veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Trabalhos de transcrição e selecione Criar tarefa (canto superior direito). Isso abre a página Especificar detalhes do trabalho.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

► **Additional settings**

3. Preencha os campos que você deseja incluir na página Especificar detalhes do trabalho e selecione Avançar. Isso leva você à página Configurar tarefa - opcional.

No painel Configurações de áudio, selecione Identificação do canal (sob o título “Tipo de identificação de áudio”).

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or partition speakers in the input audio.

Audio identification type

Channel identification

Speaker partitioning

Alternative results [Info](#)
Enable to view more transcription results

4. Selecione Criar tarefa para executar sua tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#). Para obter mais informações, consulte [StartTranscriptionJob](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--settings ChannelIdentification=true
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que permite a identificação do canal com essa tarefa.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-transcription-job.json
```

O arquivo `my-first-transcription-job.json` contém o seguinte corpo da solicitação.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "LanguageCode": "en-US",
  "Settings": {
    "ChannelIdentification": true
  }
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para identificar canais usando o método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Settings = {
        'ChannelIdentification': True
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
```

```
print(status)
```

Usando a identificação de canais em uma transcrição de streaming

Para identificar canais em uma transcrição de streaming, você pode usar HTTP/2 ou WebSockets; veja exemplos a seguir:

Fluxo do HTTP/2

Este exemplo cria uma solicitação HTTP/2 que separa canais em sua saída de transcrição. Para obter mais informações sobre como usar o streaming HTTP/2 com Amazon Transcribe, consulte [Configurar um fluxo de HTTP/2](#). Para obter mais detalhes sobre parâmetros e cabeçalhos específicos para Amazon Transcribe, consulte [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-channel-identification: TRUE
transfer-encoding: chunked
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

WebSocket riacho

Este exemplo cria um URL pré-assinado que separa os canais em sua saída de transcrição. Quebras de linha foram adicionadas para uma melhor leitura. Para obter mais informações sobre como usar WebSocket streams com Amazon Transcribe, consulte [Configurando um WebSocket stream](#). Para obter mais detalhes sobre os parâmetros, consulte [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
```



```
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&language-code=en-US
&specialty=PRIMARYCARE
&type=DICTATION
&media-encoding=flac
&sample-rate=16000
&channel-identification=TRUE
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Exemplo de saída de identificação de canal (lote)

Aqui está um exemplo de saída para uma transcrição em lote com identificação de canal ativada.

```
{
  "jobName": "my-first-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "I've been on hold for an hour. Sorry about that."
      }
    ],
    "channel_labels": {
      "channels": [
        {
          "channel_label": "ch_0",
          "items": [
            {
              "channel_label": "ch_0",
              "start_time": "4.86",
              "end_time": "5.01",
              "alternatives": [
                {
                  "confidence": "1.0",
```

```
        "content": "I've"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "start_time": "5.01",
    "end_time": "5.16",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "been"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "start_time": "5.16",
    "end_time": "5.28",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "on"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "start_time": "5.28",
    "end_time": "5.62",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "hold"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "start_time": "5.62",
```

```
        "end_time": "5.83",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "for"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "6.1",
        "end_time": "6.25",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "an"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "6.25",
        "end_time": "6.87",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "hour"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "language_code": "en-US",
        "alternatives": [
            {
                "confidence": "0.0",
                "content": "."
            }
        ],
        "type": "punctuation"
    }
}
```

```
    ]
  },
  {
    "channel_label": "ch_1",
    "items": [
      {
        "channel_label": "ch_1",
        "start_time": "8.5",
        "end_time": "8.89",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "Sorry"
          }
        ],
        "type": "pronunciation"
      },
      {
        "channel_label": "ch_1",
        "start_time": "8.89",
        "end_time": "9.06",
        "alternatives": [
          {
            "confidence": "0.9176",
            "content": "about"
          }
        ],
        "type": "pronunciation"
      },
      {
        "channel_label": "ch_1",
        "start_time": "9.06",
        "end_time": "9.25",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "that"
          }
        ],
        "type": "pronunciation"
      },
      {
        "channel_label": "ch_1",
        "alternatives": [
```

```
        {
            "confidence": "0.0",
            "content": "."
        }
    ],
    "type": "punctuation"
}
]
}
],
"number_of_channels": 2
},
"items": [
    {
        "channel_label": "ch_0",
        "start_time": "4.86",
        "end_time": "5.01",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "I've"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "5.01",
        "end_time": "5.16",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "been"
            }
        ],
        "type": "pronunciation"
    },
    {
        "channel_label": "ch_0",
        "start_time": "5.16",
        "end_time": "5.28",
        "alternatives": [
            {
                "confidence": "1.0",
```

```
        "content": "on"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "start_time": "5.28",
    "end_time": "5.62",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "hold"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "start_time": "5.62",
    "end_time": "5.83",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "for"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "start_time": "6.1",
    "end_time": "6.25",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "an"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "start_time": "6.25",
```

```
    "end_time": "6.87",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "hour"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_0",
    "alternatives": [
      {
        "confidence": "0.0",
        "content": "."
      }
    ],
    "type": "punctuation"
  },
  {
    "channel_label": "ch_1",
    "start_time": "8.5",
    "end_time": "8.89",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "Sorry"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_1",
    "start_time": "8.89",
    "end_time": "9.06",
    "alternatives": [
      {
        "confidence": "0.9176",
        "content": "about"
      }
    ],
    "type": "pronunciation"
  },
  {
```

```
    "channel_label": "ch_1",
    "start_time": "9.06",
    "end_time": "9.25",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "that"
      }
    ],
    "type": "pronunciation"
  },
  {
    "channel_label": "ch_1",
    "alternatives": [
      {
        "confidence": "0.0",
        "content": "."
      }
    ],
    "type": "punctuation"
  }
]
},
"status": "COMPLETED"
}
```


Identificar os idiomas dominantes na mídia

O Amazon Transcribe é capaz de identificar automaticamente os idiomas falados na mídia sem que você precise especificar um código de idioma.

A [identificação do idioma em lote](#) pode identificar o idioma dominante falado no arquivo de mídia ou, se a mídia contiver vários idiomas, poderá identificar todos os idiomas falados. Para melhorar a precisão da identificação do idioma, você pode, opcionalmente, fornecer uma lista de dois ou mais idiomas que você acha que podem estar presentes na mídia.

A [identificação do idioma de streaming](#) pode identificar um idioma por canal (há suporte para no máximo dois canais) ou, se o streaming tiver vários idiomas, poderá identificar todos os idiomas falados. As solicitações de streaming devem ter no mínimo duas opções de idioma adicionais incluídas na solicitação. Fornecer opções de idioma permite uma identificação mais rápida do idioma. Quanto mais rápido o Amazon Transcribe conseguir identificar o idioma, menor será a alteração na perda de dados nos primeiros segundos do streaming.

Important

As transcrições de streaming e em lote são compatíveis com diferentes idiomas. Consulte detalhes na coluna Entrada de dados da [tabela de idiomas compatíveis](#). Observe que, no momento, o sueco e o vietnamita não são compatíveis com a identificação de idioma.

Para saber mais sobre monitoramento e eventos com identificação de idioma, consulte [Eventos de identificação de idiomas](#).

Identificação de idioma com trabalhos de transcrição em lote

Use a identificação de idioma em lote para identificar automaticamente um ou mais idiomas no arquivo de mídia.

Se a mídia tiver apenas um idioma, você poderá habilitar a [identificação de um único idioma](#), que identifica o idioma dominante falado no arquivo de mídia e cria a transcrição usando somente esse idioma.

Se a mídia contiver mais de um idioma, você poderá habilitar a [identificação de vários idiomas](#), que identifica todos os idiomas falados no arquivo de mídia e cria a transcrição usando cada idioma

identificado. Observe que uma transcrição de vários idiomas é produzida. Você pode usar outros serviços Amazon Translate, como traduzir sua transcrição.

Consulte na tabela de [idiomas compatíveis](#) uma lista completa dos idiomas compatíveis e os códigos de idioma associados.

Para conseguir melhores resultados, o arquivo de mídia deve ter pelo menos 30 segundos de fala.

Para exemplos de uso com o AWS Management Console, AWS CLI, e o SDK do AWS Python, consulte. [Usar a identificação de idioma com transcrições em lote](#)

Identificar idiomas em áudio com vários idiomas

A identificação de vários idiomas é destinada a arquivos de mídia com vários idiomas e fornece uma transcrição que reflete todos os [idiomas compatíveis](#) falados na mídia. Isso significa que, se os locutores mudarem de idioma no meio da conversa ou se cada participante falar um idioma diferente, a saída de transcrição detectará e transcreverá cada idioma corretamente. Por exemplo, se a mídia contiver um locutor bilíngue que esteja alternando entre inglês americano (en-US) e hindi (hi-IN), a identificação de vários idiomas poderá identificar e transcrever o inglês americano falado como en-US e o hindi falado como hi-IN.

Isso difere da identificação de um único idioma, em que apenas um idioma dominante é usado para criar uma transcrição. Nesse caso, qualquer idioma falado que não seja o dominante é transcrito incorretamente.

Note

No momento, não é possível usar edição e modelos de idioma personalizados com a identificação de vários idiomas.

Note

Atualmente, os seguintes idiomas são suportados com identificação em vários idiomas: en-AB, en-AU, en-GB, en-IE, en-IN, en-NZ, en-US, en-WL, en-ZA, es-ES, es-US, fr-CA, fr-FR, zh-CN, zh-TW, pt-BR, pt-PT, de-CH, de-DE, af-za, ar-AE, da-DK, He-il, Hi-in, Identificação, Fa-ir, It-it, Ja-jp, Ko-KR, MS-my, NL-nl, Ru-ru, Ta-in, Te-in, Th-th, Tr-tr

As transcrições de vários idiomas fornecem um resumo dos idiomas detectados e o tempo total em que cada idioma é falado na mídia. Veja um exemplo abaixo:

```
"results": {
  "transcripts": [
    {
      "transcript": "welcome to Amazon transcribe. ## ## ##### ### #### ####
## #### ## #####"
    }
  ],
  ...
  "language_codes": [
    {
      "language_code": "en-US",
      "duration_in_seconds": 2.45
    },
    {
      "language_code": "hi-IN",
      "duration_in_seconds": 5.325
    },
    {
      "language_code": "ja-JP",
      "duration_in_seconds": 4.15
    }
  ]
}
```

Melhorar a precisão da identificação de idiomas

Com a identificação de idioma, você tem a opção de incluir uma lista de idiomas que você acha que podem estar presentes na mídia. A inclusão de opções de idioma (`LanguageOptions`) restringe Amazon Transcribe o uso somente dos idiomas que você especifica ao combinar seu áudio com o idioma correto, o que pode acelerar a identificação do idioma e melhorar a precisão associada à atribuição do dialeto correto do idioma.

Se você optar por incluir códigos de idioma, deverá especificar, pelo menos, dois. Não há limite para o número de códigos de idioma que você pode incluir, mas recomendamos usar entre dois e cinco para otimizar a eficiência e a precisão.

Note

Se você incluir códigos de idioma em sua solicitação e nenhum dos códigos de idioma fornecidos corresponder ao idioma ou idiomas identificados em seu áudio, Amazon Transcribe selecionará a correspondência de idioma mais próxima dos códigos de idioma especificados. Em seguida, ele produz uma transcrição nesse idioma. Por exemplo, se sua mídia estiver em inglês dos EUA (en-US) e você Amazon Transcribe fornecer os códigos de idiomas zh-CN, e fr-FR de-DE, Amazon Transcribe provavelmente corresponderá sua mídia ao alemão (de-DE) e produzirá uma transcrição em alemão. A incompatibilidade entre códigos de idioma e idiomas falados pode resultar em uma transcrição imprecisa; portanto, recomendamos cautela ao incluir códigos de idioma.

Utilizar a identificação de idioma com outros recursos do Amazon Transcribe

Você pode usar a identificação de idioma em lote com qualquer outro recurso do Amazon Transcribe. Ao combinar a identificação de idioma com outros recursos, você se limitará aos idiomas compatíveis com esses recursos. Por exemplo, se estiver usando a identificação de idioma com a redação de conteúdo, você estará limitado ao inglês dos EUA (en-US) ou ao espanhol dos EUA (es-US), pois esse é o único idioma disponível para redação. Consulte [Idiomas oferecidos e recursos específicos do idioma](#) para obter mais informações.

Important

Se você estiver usando a identificação automática de idioma com a redação de conteúdo ativada e seu áudio contiver outros idiomas além do inglês dos EUA (en-US) ou do espanhol dos EUA (es-US), somente o conteúdo em inglês dos EUA ou espanhol dos EUA será redigido na sua transcrição. Outros idiomas não podem ser editados e não há avisos ou falhas no trabalho.

Modelos de idioma personalizados, vocabulários personalizados e filtros de vocabulário personalizados

Se você quiser adicionar um ou mais modelos de idioma personalizados, vocabulários personalizados ou filtros de vocabulário personalizados à solicitação de identificação de idioma,

inclua o parâmetro [LanguageIdSettings](#). Depois, é possível especificar um código de idioma com um modelo de idioma personalizado, vocabulário personalizado e filtro de vocabulário personalizado correspondentes. Observe que a identificação de vários idiomas não é compatível com modelos de idioma personalizados.

É recomendável que você inclua `LanguageOptions` ao usar [LanguageIdSettings](#) para garantir que o dialeto correto do idioma seja identificado. Por exemplo, se você especificar um vocabulário en-US personalizado, mas Amazon Transcribe determinar qual é o idioma falado em sua mídia en-AU, seu vocabulário personalizado não será aplicado à sua transcrição. Se você incluir `LanguageOptions` e especificar en-US como o único dialeto do idioma inglês, o vocabulário personalizado será aplicado à transcrição.

Para ver exemplos de [LanguageIdSettings](#) em uma solicitação, consulte a Opção 2 da AWS CLI e os painéis suspensos de SDKs da AWS na seção [Usar a identificação de idioma com transcrições em lote](#).

Usar a identificação de idioma com transcrições em lote

Você pode usar a identificação automática de idioma em um trabalho de transcrição em lote usando o AWS Management Console, a AWS CLI ou os SDKs da AWS ; veja os exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Tarefas de transcrição e selecione Criar tarefa (no canto superior direito). Isso abre a página Especificar os detalhes da tarefa.
3. No painel Configurações de tarefa, encontre a seção Configurações de idioma e selecione Identificação automática de idioma ou Identificação automática de vários idiomas.

Você pode selecionar várias opções de idioma (na caixa suspensa Selecionar idiomas) caso saiba quais idiomas estão presentes no arquivo de áudio. Fornecer opções de idioma pode melhorar a precisão, mas não é obrigatório.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

- Specific language** [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.
- Automatic language identification** [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.
- Automatic multiple languages identification** [Info](#)
If there are multiple languages spoken in your audio files and you're not sure what these languages are, choose this option. This selection provides limited additional processing options compared to **Specific language**.

Language options for automatic language identification - *optional*

To improve accuracy, choose at least two languages spoken the most often in your audio library. Amazon Transcribe chooses from one of the languages you've specified to transcribe each audio file. Leave this field empty if you're unsure about which languages to select.

- English, US (en-US)
- English, AU (en-AU)
- English, UK (en-GB)
- Hindi, IN (hi-IN)
- Spanish, US (es-US)

4. Preencha os outros campos que deseja incluir na página Especificar os detalhes da tarefa e selecione Próximo. Isso leva você à página Configurar tarefa - opcional.

Configure job - *optional* [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Cancel Previous Create Job

5. Selecione Criar tarefa para executar a tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o `IdentifyLanguage` parâmetro. Para obter mais informações, consulte [StartTranscriptionJob](#) e [LanguageIdSettings](#).

Opção 1: sem o parâmetro `language-id-settings`. Use essa opção se você não estiver incluindo um modelo de idioma personalizado, vocabulário personalizado ou filtro de vocabulário personalizado em na solicitação. `language-options` é opcional, mas recomendado.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--identify-language \ (or --identify-multiple-languages) \  
--language-options "en-US" "hi-IN"
```

Opção 1: com o parâmetro `language-id-settings`. Use essa opção se estiver incluindo um modelo de idioma personalizado, um vocabulário personalizado ou um filtro de vocabulário personalizado na solicitação.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--identify-language \ (or --identify-multiple-languages) \  
--language-options "en-US" "hi-IN" \  
--language-id-settings en-US=VocabularyName=my-en-US-vocabulary,en-  
US=VocabularyFilterName=my-en-US-vocabulary-filter,en-US=LanguageModelName=my-en-US-  
language-model,hi-IN=VocabularyName=my-hi-IN-vocabulary,hi-IN=VocabularyFilterName=my-  
hi-IN-vocabulary-filter
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que identifica a linguagem.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-first-language-id-job.json
```

O arquivo `my-first-language-id-job.json` contém o corpo da solicitação a seguir.

Opção 1: sem o parâmetro `LanguageIdSettings`. Use essa opção se você não estiver incluindo um modelo de idioma personalizado, vocabulário personalizado ou filtro de vocabulário personalizado em na solicitação. `LanguageOptions` é opcional, mas recomendado.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "IdentifyLanguage": true, (or "IdentifyMultipleLanguages": true),
  "LanguageOptions": [
    "en-US", "hi-IN"
  ]
}
```

Opção 1: com o parâmetro `LanguageIdSettings`. Use essa opção se estiver incluindo um modelo de idioma personalizado, um vocabulário personalizado ou um filtro de vocabulário personalizado na solicitação.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "IdentifyLanguage": true, (or "IdentifyMultipleLanguages": true)
  "LanguageOptions": [
    "en-US", "hi-IN"
  ],
  "LanguageIdSettings": {
    "en-US" : {
      "LanguageModelName": "my-en-US-language-model",
      "VocabularyFilterName": "my-en-US-vocabulary-filter",
      "VocabularyName": "my-en-US-vocabulary"
    },
    "hi-IN": {
      "VocabularyName": "my-hi-IN-vocabulary",
      "VocabularyFilterName": "my-hi-IN-vocabulary-filter"
    }
  }
}
```

```
}  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para identificar o idioma do seu arquivo usando o `IdentifyLanguage` argumento do método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#) e [LanguageIdSettings](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

Opção 1: sem o parâmetro `LanguageIdSettings`. Use essa opção se você não estiver incluindo um modelo de idioma personalizado, vocabulário personalizado ou filtro de vocabulário personalizado em na solicitação. `LanguageOptions` é opcional, mas recomendado.

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-transcription-job"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri  
    },  
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',  
    OutputKey = 'my-output-files/',  
    MediaFormat = 'flac',  
    IdentifyLanguage = True, (or IdentifyMultipleLanguages = True),  
    LanguageOptions = [  
        'en-US', 'hi-IN'  
    ]  
)  
  
while True:  
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)  
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:  
        break  
    print("Not ready yet...")  
    time.sleep(5)
```

```
print(status)
```

Opção 1: com o parâmetro `LanguageIdSettings`. Use essa opção se estiver incluindo um modelo de idioma personalizado, um vocabulário personalizado ou um filtro de vocabulário personalizado na solicitação.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    MediaFormat='flac',
    IdentifyLanguage=True, (or IdentifyMultipleLanguages=True)
    LanguageOptions = [
        'en-US', 'hi-IN'
    ],
    LanguageIdSettings={
        'en-US': {
            'VocabularyName': 'my-en-US-vocabulary',
            'VocabularyFilterName': 'my-en-US-vocabulary-filter',
            'LanguageModelName': 'my-en-US-language-model'
        },
        'hi-IN': {
            'VocabularyName': 'my-hi-IN-vocabulary',
            'VocabularyFilterName': 'my-hi-IN-vocabulary-filter'
        }
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
```

```
print(status)
```

Identificação de idioma com transcrições de streaming

A identificação do idioma de streaming pode identificar o idioma dominante falado no streaming de mídia. O Amazon Transcribe requer um mínimo de três segundos de fala para identificar o idioma.

Se o streaming tiver apenas um idioma, você poderá habilitar a identificação de um único idioma, que identifica o idioma dominante falado no arquivo de mídia e cria a transcrição usando somente esse idioma.

Se o streaming tiver mais de um idioma, você poderá habilitar a identificação de vários idiomas, que identifica todos os idiomas falados no streaming e cria a transcrição usando cada idioma identificado. Observe que uma transcrição de vários idiomas é produzida. É possível usar outros serviços, como o Amazon Transcribe, para traduzir a transcrição.

Para usar a identificação do idioma de streaming, você deve fornecer pelo menos dois códigos de idioma e pode selecionar somente um dialeto de idioma, por idioma e por streaming. Isso significa que você não pode selecionar en-US e en-AU como opções de idioma para a mesma transcrição.

Você também tem a opção de selecionar um idioma preferido no conjunto de códigos de idioma fornecido. Adicionar um idioma preferencial pode acelerar o processo de identificação do idioma, o que é útil para clipes de áudio curtos.

Important

Se nenhum dos códigos de idioma fornecidos corresponder ao idioma ou idiomas identificados no áudio, o Amazon Transcribe selecionará o idioma mais próximo entre os códigos de idioma especificados. Depois, ele produz uma transcrição nesse idioma. Por exemplo, se a mídia estiver em inglês dos EUA (en-US) e você Amazon Transcribe fornecer os códigos de idioma zh-CN, fr-FR e de-DE, o Amazon Transcribe provavelmente encontrará uma correspondência da mídia com o alemão (de-DE) e produzirá uma transcrição nesse idioma. A incompatibilidade entre códigos de idioma e os idiomas falados pode resultar em uma transcrição imprecisa, portanto, recomendamos cautela ao incluir códigos de idioma.

Se a mídia tiver dois canais, o Amazon Transcribe poderá identificar o idioma dominante falado em cada canal. Nesse caso, defina o parâmetro [ChannelIdentification](#) como `true` e cada canal

será transcrito separadamente. Observe que o padrão para esse parâmetro é `false`. Se você não alterar, apenas o primeiro canal será transcrito e somente um idioma será identificado.

A identificação do idioma de streaming não pode ser combinada com modelos de idioma personalizados ou edição. Se combinar a identificação de idioma com outros recursos, você se limitará aos idiomas compatíveis com esses recursos e também com transcrições de streaming. Consulte [Idiomas compatíveis](#).

Note

PCM e FLAC são os únicos formatos de áudio compatíveis para identificação de idiomas de streaming.

Identificação de idiomas em áudio com vários idiomas

A identificação de vários idiomas é destinada a streamings com vários idiomas e fornece uma transcrição que reflete todos os idiomas compatíveis falados no streaming. Isso significa que, se os locutores mudarem de idioma no meio da conversa ou se cada participante falar um idioma diferente, a saída de transcrição detectará e transcreverá cada idioma corretamente.

Por exemplo, se o streaming tiver um locutor bilíngue que alterne entre inglês dos EUA (`en-US`) e hindi (`hi-IN`), a identificação de vários idiomas poderá identificar e transcrever o inglês dos EUA falado como `en-US` e o hindi falado como `hi-IN`. Isso difere da identificação de um único idioma, em que apenas um idioma dominante é usado para criar uma transcrição. Nesse caso, qualquer idioma falado que não seja o dominante é transcrito incorretamente.

Note

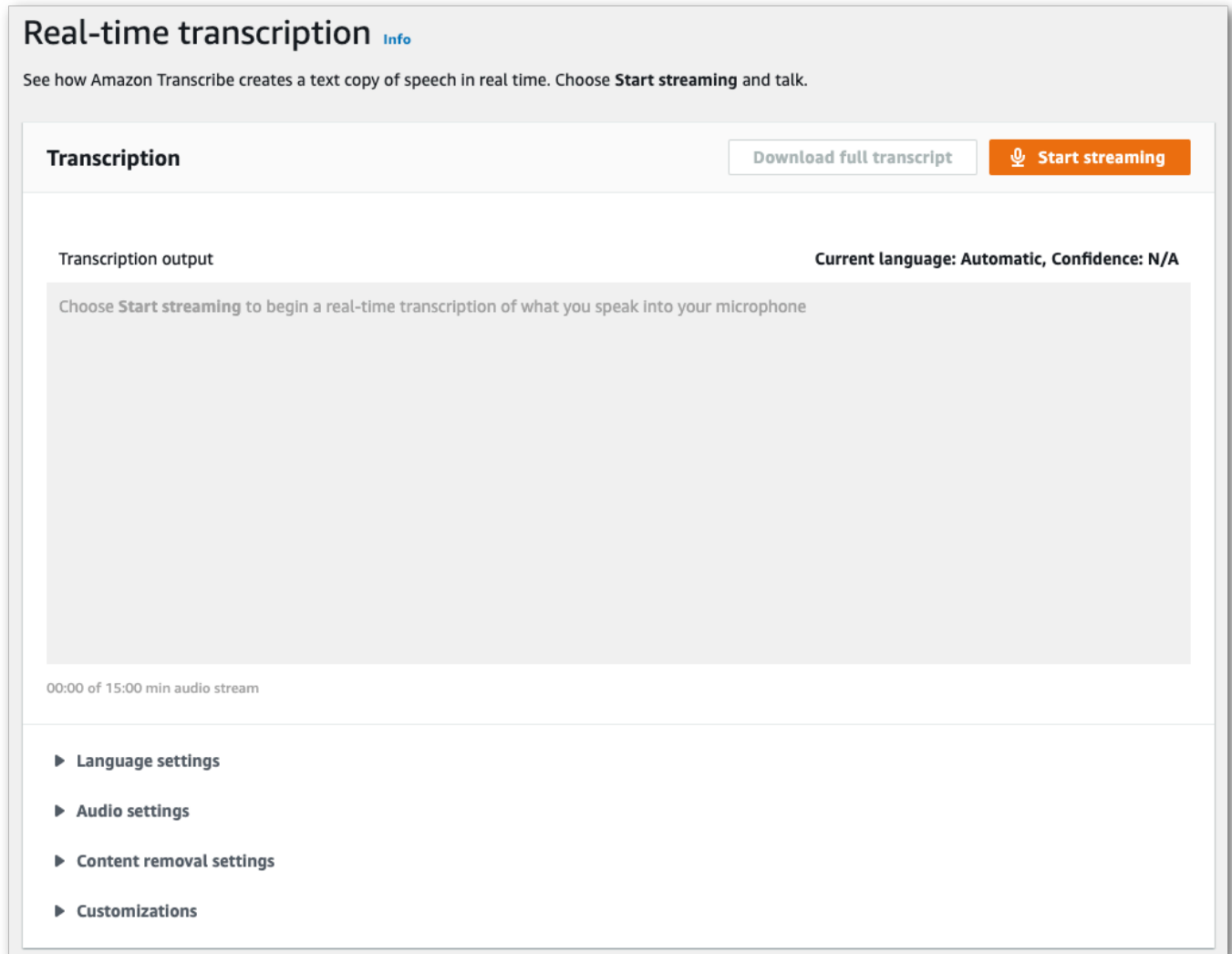
Atualmente, os modelos de edição e de linguagem personalizada não são compatíveis com a identificação de vários idiomas.

Usar a identificação de idioma com streaming de mídia

É possível usar a identificação automática de idioma em uma transcrição de streaming usando o AWS Management Console, HTTP/2 ou WebSockets; veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Transcrição em tempo real. Role para baixo até Configurações de idioma e expanda esse campo se ele estiver minimizado.



3. Selecione Identificação automática de idioma ou Identificação automática de vários idiomas.

▼ Language settings

Language settings

You can select a specific language for your transcription or have Amazon Transcribe identify the predominant language in your media and perform the transcription in that language.

Specific language

If you know the language spoken in your source audio, choose this option to get the most accurate results.

Automatic language identification [Info](#)

If you don't know the language spoken in your audio files, choose this option.

Automatic multiple languages identification [Info](#)

If there are multiple languages spoken in your audio files and you're not sure what these languages are, choose this option. This selection provides limited additional processing options compared to **Specific language**.

Language options for automatic language identification

To improve language identification accuracy, select a minimum of 2 language options.

Choose language(s) ▼

Preferred language - *optional*

Specify one preferred language from your previous selection.

Choose language ▼

▶ **Audio settings**

▶ **Content removal settings**

▶ **Customizations**

4. Forneça no mínimo dois códigos de idioma para a transcrição. Observe que você pode fornecer somente um dialeto por idioma. Por exemplo, você não pode selecionar en-US e fr-CA como opções de idioma para a mesma transcrição.

▼ **Language settings**

Language settings
You can select a specific language for your transcription or have Amazon Transcribe identify the predominant language in your media and perform the transcription in that language.

Specific language
If you know the language spoken in your source audio, choose this option to get the most accurate results.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option.

Automatic multiple languages identification [Info](#)
If there are multiple languages spoken in your audio files and you're not sure what these languages are, choose this option. This selection provides limited additional processing options compared to **Specific language**.

Language options for automatic language identification
To improve language identification accuracy, select a minimum of 2 language options.

Choose language(s) ▼

English, US (en-US) ✕ French, CA (fr-CA) ✕

Preferred language - optional
Specify one preferred language from your previous selection.

Choose language ▲

Q

None
English, US (en-US)
French, CA (fr-CA)

5. (Opcional) No subconjunto de idiomas que você selecionou na etapa anterior, é possível escolher um idioma preferido para a transcrição.

▼ **Language settings**

Language settings
 You can select a specific language for your transcription or have Amazon Transcribe identify the predominant language in your media and perform the transcription in that language.

Specific language
 If you know the language spoken in your source audio, choose this option to get the most accurate results.

Automatic language identification [Info](#)
 If you don't know the language spoken in your audio files, choose this option.

Language options for automatic language identification
 To improve language identification accuracy, select a minimum of 2 language options.

Choose language(s) ▼

English, US (en-US) X French, CA (fr-CA) X

Preferred language - optional
 Specify one preferred language from your previous selection.

Choose language ▲

None

English, US (en-US)

French, CA (fr-CA)

► **Customizations**

6. Agora está tudo pronto para transcrever o streaming. Selecione Iniciar streaming e comece a falar. Para finalizar o ditado, selecione Interromper transmissão.

Streaming de HTTP/2

Este exemplo cria uma solicitação HTTP/2 com a identificação de idioma habilitada. Consulte mais informações sobre como usar o streaming HTTP/2 com o Amazon Transcribe em [Configurar um fluxo de HTTP/2](#). Consulte mais detalhes sobre parâmetros e cabeçalhos específicos do Amazon Transcribe em [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
```

```
x-amzn-transcribe-identify-language: true
x-amzn-transcribe-language-options: en-US,de-DE
x-amzn-transcribe-preferred-language: en-US
transfer-encoding: chunked
```

Este exemplo cria uma solicitação HTTP/2 com a identificação de vários idiomas habilitada. Consulte mais informações sobre como usar o streaming HTTP/2 com o Amazon Transcribe em [Configurar um fluxo de HTTP/2](#). Consulte mais detalhes sobre parâmetros e cabeçalhos específicos do Amazon Transcribe em [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-identify-multiple-languages: true
x-amzn-transcribe-language-options: en-US,de-DE
x-amzn-transcribe-preferred-language: en-US
transfer-encoding: chunked
```

Se você usar `identify-language` ou `identify-multiple-languages` na solicitação, também deverá incluir `language-options`. Você não pode usar `language-code` e `identify-language` na mesma solicitação.

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações da AWS API estão listados na seção [Parâmetros comuns](#).

Streaming do WebSocket

Este exemplo cria um URL pré-assinado que usa identificação de idioma em um streaming do WebSocket. Quebras de linha foram adicionadas para uma melhor leitura. Consulte mais informações sobre o uso de streamings do WebSocket com o Amazon Transcribe em [Configurando um WebSocket stream](#). Consulte mais detalhes sobre os parâmetros em [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&media-encoding=flac
&sample-rate=16000
&identify-language=true
&language-options=en-US,de-DE
&preferred-language=en-US
```

Este exemplo cria um URL pré-assinado que usa identificação de vários idiomas em um streaming do WebSocket. Quebras de linha foram adicionadas para uma melhor leitura. Consulte mais informações sobre o uso de streamings do WebSocket com o Amazon Transcribe em [Configurando um WebSocket stream](#). Consulte mais detalhes sobre os parâmetros em [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&media-encoding=flac
&sample-rate=16000
&identify-multiple-languages=true
&language-options=en-US,de-DE
&preferred-language=en-US
```

Se você usar `identify-language` ou `identify-multiple-languages` na solicitação, também deverá incluir `language-options`. Você não pode usar `language-code` e `identify-language` na mesma solicitação.

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações da AWS API estão listados na seção [Parâmetros comuns](#).

Transcrições alternativas

Quando Amazon Transcribe transcreve o áudio, ele cria versões diferentes da mesma transcrição e atribui uma pontuação de confiança a cada versão. Em uma transcrição típica, você só obtém a versão com a maior pontuação de confiança.

Se você ativar as transcrições alternativas, Amazon Transcribe retornará outras versões da sua transcrição que tenham níveis de confiança mais baixos. Você pode optar por ter até 10 transcrições alternativas retornadas. Se você especificar um número maior de alternativas do que o Amazon Transcribe identificador, somente o número real de alternativas será retornado.

Todas as alternativas estão localizadas no mesmo arquivo de saída da transcrição e são apresentadas no nível do segmento. Os segmentos são pausas naturais na fala, como uma mudança no alto-falante ou uma pausa no áudio.

As transcrições alternativas só estão disponíveis para transcrições em lote.

Sua maneira a seguir é estruturada da maneira a seguir. As elipses (. . .) nos exemplos de código indicam onde o conteúdo foi removido por questões de brevidade.

1. Uma transcrição final completa para um determinado segmento.

```
"results": {
  "language_code": "en-US",
  "transcripts": [
    {
      "transcript": "The amazon is the largest rainforest on the planet."
    }
  ],
  . . .
}
```

2. Uma pontuação de confiança para cada palavra na `transcript` seção anterior.

```
"items": [
  {
    "start_time": "1.15",
    "end_time": "1.35",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "The"
      }
    ]
  }
  . . .
]
```

```

    ],
    "type": "pronunciation"
  },
  {
    "start_time": "1.35",
    "end_time": "2.05",
    "alternatives": [
      {
        "confidence": "1.0",
        "content": "amazon"
      }
    ],
    "type": "pronunciation"
  },

```

3. Suas transcrições alternativas estão localizadas na `segments` parte da saída da transcrição. As alternativas para cada segmento são ordenadas por pontuação de confiança decrescente.

```

"segments": [
  {
    "start_time": "1.04",
    "end_time": "5.065",
    "alternatives": [
      ...
      "transcript": "The amazon is the largest rain forest on the
planet.",
      "items": [
        {
          "start_time": "1.15",
          "confidence": "1.0",
          "end_time": "1.35",
          "type": "pronunciation",
          "content": "The"
        },
        ...
        {
          "start_time": "3.06",
          "confidence": "0.0037",
          "end_time": "3.38",
          "type": "pronunciation",
          "content": "rain"
        }
      ],

```

```
{
  "start_time": "3.38",
  "confidence": "0.0037",
  "end_time": "3.96",
  "type": "pronunciation",
  "content": "forest"
},
```

4. Um status no final da saída da transcrição.

```
"status": "COMPLETED"
}
```

Solicitando transcrições alternativas

Você pode solicitar transcrições alternativas usando os SDKs AWS Management ConsoleAWS CLI, ou AWSos SDKs; veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Trabalhos de transcrição e selecione Criar tarefa (canto superior direito). Isso abre a página Especificar detalhes do trabalho.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

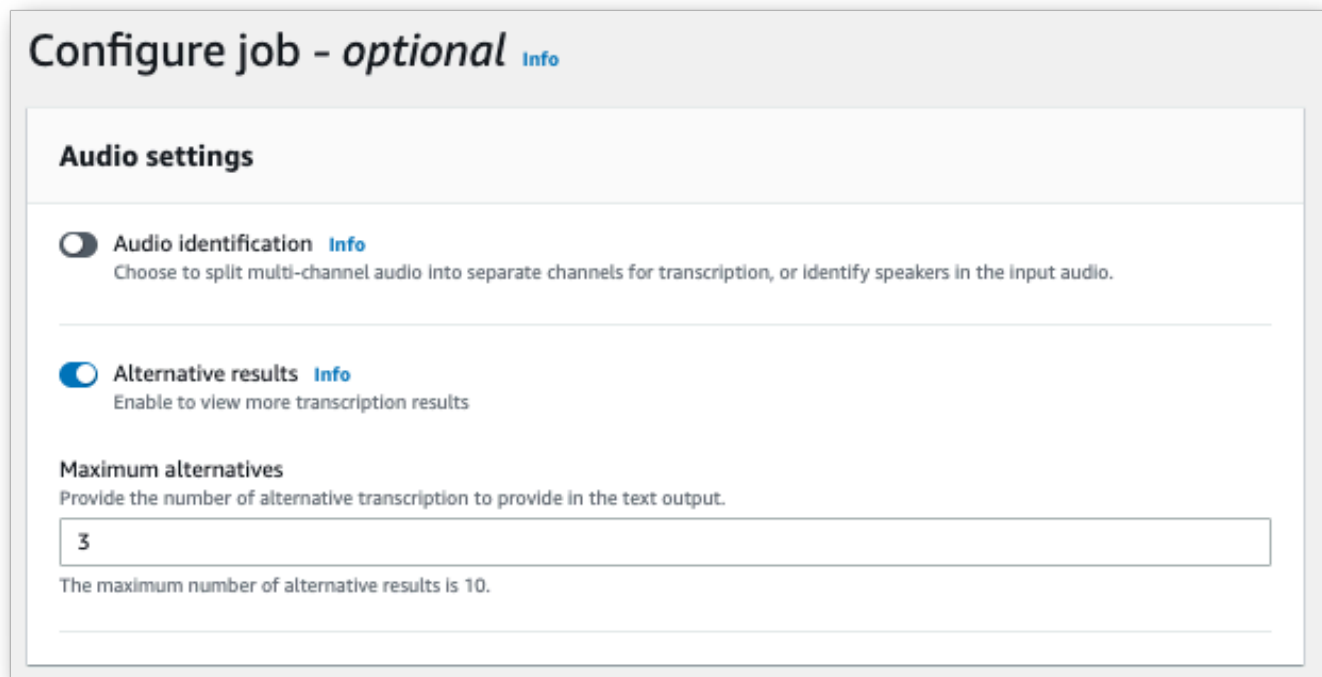
Language

Choose the language of the input audio.

► **Additional settings**

3. Preencha os campos que você deseja incluir na página Especificar detalhes do trabalho e selecione Avançar. Isso leva você à página Configurar tarefa - opcional.

Selecione Resultados alternativos e especifique o número máximo de resultados de transcrição alternativos que você deseja em sua transcrição.



4. Selecione Criar tarefa para executar sua tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o `ShowAlternatives` parâmetro. Para obter mais informações, consulte [StartTranscriptionJob](#) e [ShowAlternatives](#).

Observe que, se você incluir `ShowAlternatives=true` em sua solicitação, também deverá incluir `MaxAlternatives`.

```
aws transcribe start-transcription-job \
--region us-west-2 \
--transcription-job-name my-first-transcription-job \
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \
--output-bucket-name DOC-EXAMPLE-BUCKET \
--output-key my-output-files/ \
--language-code en-US \
--settings ShowAlternatives=true,MaxAlternatives=4
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que inclui transcrições alternativas.

```
aws transcribe start-transcription-job \
```

```
--region us-west-2 \  
--cli-input-json file://filepath/my-first-alt-transcription-job.json
```

O arquivo `my-first-alt-transcription-job.json` contém o seguinte corpo da solicitação.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 4  
  }  
}
```

AWS SDK for Python (Boto3)

O exemplo a seguir usa o AWS SDK for Python (Boto3) para solicitar transcrições alternativas usando o `ShowAlternatives` argumento para o método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#) e [ShowAlternatives](#).

Para obter exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e serviços cruzados, consulte o [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#) capítulo.

Observe que, se você incluir `ShowAlternatives': True` em sua solicitação, também deverá incluir `MaxAlternatives`.

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-transcription-job"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri
```

```
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Settings = {
        'ShowAlternatives':True,
        'MaxAlternatives':4
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Melhorando a precisão da transcrição com vocabulários e modelos de linguagem personalizados

Se sua mídia contiver termos específicos de domínio ou não padronizados, como nomes de marcas, acrônimos, palavras técnicas e jargões, Amazon Transcribe talvez não capte corretamente esses termos em sua saída de transcrição.

Para corrigir imprecisões de transcrição e personalizar sua saída para seu caso de uso específico, você pode criar [Vocabulários personalizados](#) [Modelos de linguagem personalizados](#) e.

- [Vocabulários personalizados](#) são projetados para ajustar e impulsionar o reconhecimento e a formatação de palavras específicas em todos os contextos. Isso envolve o fornecimento Amazon Transcribe de palavras e, opcionalmente, formulários de pronúncia e exibição.

Se não Amazon Transcribe estiver renderizando corretamente termos específicos em suas transcrições, você pode criar um arquivo de vocabulário personalizado que informa Amazon Transcribe como você deseja que esses termos sejam exibidos. Essa abordagem específica de palavras é mais apropriada para corrigir termos como nomes de marcas e acrônimos.

- [Modelos de linguagem personalizados](#) são projetados para capturar o contexto associado aos termos. Isso envolve o fornecimento Amazon Transcribe de um grande volume de dados de texto específicos do domínio.

Se não Amazon Transcribe estiver renderizando corretamente os termos técnicos ou estiver usando o homófono incorreto em suas transcrições, você pode criar um modelo de linguagem personalizado que ensine Amazon Transcribe o idioma específico do seu domínio. Por exemplo, um modelo de linguagem personalizado pode aprender quando usar 'floe' (bloco de gelo) versus 'flow' (fluxo linear).

Essa abordagem sensível ao contexto é mais apropriada para transcrever grandes volumes de discursos específicos de um domínio. Modelos de linguagem personalizados podem produzir melhorias significativas na precisão apenas em relação aos vocabulários personalizados. Ao usar transcrições em lote, você pode incluir um modelo de idioma personalizado e um vocabulário personalizado em sua solicitação.

i Tip

Para obter a maior precisão de transcrição, use vocabulários personalizados em conjunto com seus modelos de linguagem personalizados.

Para ver uma demonstração em vídeo sobre como criar um vocabulário personalizado usando o AWS Management Console, consulte [Usando um vocabulário personalizado](#).

Para uma demonstração em vídeo sobre como criar e usar modelos de linguagem personalizados, consulte [Usando modelos de linguagem personalizados \(CLM\) para aumentar a precisão da transcrição](#).

i Mergulhe mais fundo com o blog AWS Machine Learning

Vocabulários do vocabulário:

- [Transcrições ao vivo de corridas de F1 usando Amazon Transcribe](#)

Modelos de linguagem personalizados:

- [Criação de modelos de linguagem personalizados para aumentar o speech-to-text desempenho Amazon Transcribe](#)
- [Aumente a precisão da transcrição das palestras em sala de aula com modelos de linguagem personalizados para Amazon Transcribe](#)

Vocabulários personalizados

Use vocabulários personalizados para melhorar a precisão da transcrição de uma ou mais palavras específicas. Geralmente, são termos específicos do domínio, como nomes de marcas e acrônimos, nomes próprios e palavras que o Amazon Transcribe não está processando corretamente.

Os vocabulários personalizados podem ser usados com todos os idiomas oferecidos. Observe que somente os caracteres listados no [conjunto de caracteres](#) do seu idioma podem ser usados em um vocabulário personalizado.

⚠ Important

Você é responsável pela integridade de seus dados ao usar o Amazon Transcribe. Não insira informações confidenciais, informações de identificação pessoal (PII) ou informações de saúde protegidas (PHI) em um vocabulário personalizado.

Considerações ao criar um vocabulário personalizado:

- Você pode ter até 100 arquivos de vocabulário personalizados por Conta da AWS
- O limite de tamanho para cada arquivo de vocabulário personalizado é de 50 Kb
- Se estiver usando a API para criar seu vocabulário personalizado, o respectivo arquivo deverá estar no formato de texto (*.txt). Se estiver usando o AWS Management Console, seu arquivo de vocabulário pode estar no formato de texto (*.txt) ou no formato de valores separados por vírgula (*.csv).
- Cada entrada em um vocabulário personalizado não pode ultrapassar 256 caracteres
- Para usar um vocabulário personalizado, ele deve ter sido criado da Região da AWS mesma forma que sua transcrição.

ℹ Tip

Você pode testar seu vocabulário personalizado usando o AWS Management Console. Quando seu vocabulário personalizado estiver pronto para uso, faça login no AWS Management Console, selecione Transcrição em tempo real, vá até Personalizações, ative Vocabulário personalizado e selecione seu vocabulário personalizado na lista suspensa. Em seguida, selecione Iniciar streaming. Fale algumas palavras do vocabulário personalizado via microfone para ver se elas são processadas corretamente.

Tabelas de vocabulário personalizadas versus listas

⚠ Important

Os vocabulários personalizados em formato de lista estão sendo descontinuados. Se você estiver criando um vocabulário personalizado, use o [formato de tabela](#).


As tabelas oferecem mais opções e maior controle sobre a entrada e saída de palavras no vocabulário personalizado. Com tabelas, você deve especificar várias categorias (Phrase and DisplayAs), permitindo que você ajuste a saída.

As listas não têm opções adicionais, então você só pode digitar as entradas conforme quiser que elas apareçam na transcrição, substituindo todos os espaços por hifens.

Todos os AWS SDKs AWS Management Console AWS CLI,, e usam tabelas de vocabulário personalizadas da mesma forma; as listas são usadas de forma diferente para cada método e, portanto, podem exigir formatação adicional para o uso bem-sucedido entre os métodos.

Para obter mais informações, consulte [Criar um vocabulário personalizado usando uma tabela](#) e [Criação de um vocabulário personalizado usando uma lista](#).

Para se aprofundar um pouco mais e saber como usar o Amazon Augmented AI com vocabulários personalizados, consulte [Start building a human review along with Amazon Transcribe](#).

 Operações de API específicas para vocabulários personalizados

[CreateVocabulary](#), [DeleteVocabulary](#), [GetVocabulary](#), [ListVocabularies](#), [UpdateVocabulary](#)

Criar um vocabulário personalizado usando uma tabela

Usar um formato de tabela é a forma preferencial de criar um vocabulário personalizado. As tabelas de vocabulário devem consistir em quatro colunas (Phrase, SoundsLike, IPA, and DisplayAs), que podem ser incluídas em qualquer ordem:

Phrase	SoundsLike	IPA	DisplayAs
Obrigatório. Cada linha na tabela deve conter uma entrada nessa coluna. Não use espaços nessa coluna.	SoundsLike não é mais compatível com o vocabulário personalizado. Por favor, deixe a coluna vazia. Todos os valores nessa coluna serão ignorados.	IPAnão é mais compatível com o vocabulário personalizado. Por favor, deixe a coluna vazia. Todos os valores nessa coluna serão ignorados. Nós	Opcional. As linhas nessa coluna podem ser deixadas vazias. Não é possível usar espaços nessa coluna.

Phrase	SoundsLike	IPA	DisplayAs
<p>Se a entrada tiver várias palavras, separe cada uma com um hífen (-). Por exemplo, o Andorra-la-Vella ou o Los-Angeles .</p> <p>Para acrônimos , todas as letras pronunciadas devem ser separadas por um ponto. O ponto final também precisa ser pronunciado. Se o acrônimo estiver no plural, você deverá usar um hífen entre ele e o “s”. Por exemplo, “CLI” é C.L.I. (não C.L.I) e “ABCs” é A.B.C.-s (não A.B.C-s).</p> <p>Se a frase consistir em uma palavra e um acrônimo, esses dois componentes devem ser separados por um hífen. Por exemplo, “DynamoDB” é Dynamo-D.B. .</p> <p>Não inclua dígitos nessa coluna; os</p>	<p>Nós removeremos o suporte para essa coluna futuramente.</p>	<p>removeremos o suporte para essa coluna futuramente.</p>	<p>Define como você deseja que a entrada apareça na saída da transcrição. Por exemplo, Andorra-la-Vella na coluna Phrase está como Andorra la Vella na coluna DisplayAs .</p> <p>Se uma linha nessa coluna estiver vazia, Amazon Transcribe usa o conteúdo da Phrase coluna para determinar a saída.</p> <p>Você pode incluir dígitos (0-9) nessa coluna.</p>

Phrase	SoundsLike	IPA	DisplayAs
números devem ser escritos por extenso. Por exemplo, “VX02Q” é V.X. - zero-two-Q. .			

Fatores a observar ao criar a tabela:

- Sua tabela deve conter os quatro cabeçalhos (Phrase, SoundsLike, IPA, and DisplayAs) das colunas. A Phrase coluna deve conter uma entrada em cada linha. A capacidade de fornecer entradas de pronúncia por meio IPA e não SoundsLike é mais suportada, e você pode deixar a coluna vazia. Todos os valores nessas colunas serão ignorados.
- Cada coluna deve ser delimitada com TAB ou vírgula (,); isso se aplica a todas as linhas do arquivo de vocabulário personalizado. Se uma linha tiver colunas vazias, você também deverá incluir um delimitador (TAB ou vírgula) para cada coluna.
- Os espaços só são permitidos nas colunas IPA e DisplayAs. Não use espaços para separar colunas.
- IPAs e não SoundsLike são mais compatíveis com o vocabulário personalizado. Por favor, deixe a coluna vazia. Todos os valores nessas colunas serão ignorados. Nós removeremos o suporte para essa coluna futuramente.
- A coluna DisplayAs aceita símbolos e caracteres especiais (por exemplo, C++). Todas as outras colunas permitem os caracteres listados na página de [conjunto de caracteres](#) do seu idioma.
- Se quiser incluir números na coluna Phrase, você deve soletrá-los. Só é possível usar dígitos (0-9) na coluna DisplayAs.
- Você deve salvar a tabela como um arquivo de texto simples (*.txt) no formato LF. Se você usar qualquer outro formato, como CRLF, seu vocabulário personalizado não poderá ser processado.
- Você deve carregar seu arquivo de vocabulário personalizado em um Amazon S3 bucket e processá-lo [CreateVocabulary](#) antes de incluí-lo em uma solicitação de transcrição. Consulte [Criar tabelas de vocabulário personalizado](#) para ver as instruções.

Note

Insira acrônimos ou outras palavras cujas letras devem ser pronunciadas individualmente como letras únicas separadas por pontos (**A.B.C.**). Para especificar a forma plural de um acrônimo, como “ABCs”, separe o “s” do acrônimo com um hífen (**A.B.C.-s**). É possível usar letras maiúsculas ou minúsculas para definir um acrônimo. Nem todos os idiomas aceitam acrônimos. Consulte [Idiomas oferecidos e recursos específicos do idioma](#).

Veja um exemplo de tabela de vocabulário personalizado (em que **[TAB]** representa um caractere de tabulação):

```
Phrase[TAB]SoundsLike[TAB]IPA[TAB]DisplayAs
Los-Angeles[TAB][TAB][TAB]Los Angeles
Eva-Maria[TAB][TAB][TAB]
A.B.C.-s[TAB][TAB][TAB]ABCs
Amazon-dot-com[TAB][TAB][TAB]Amazon.com
C.L.I.[TAB][TAB][TAB]CLI
Andorra-la-Vella[TAB][TAB][TAB]Andorra la Vella
Dynamo-D.B.[TAB][TAB][TAB]DynamoDB
V.X.-zero-two[TAB][TAB][TAB]VX02
V.X.-zero-two-Q.[TAB][TAB][TAB]VX02Q
```

Para maior clareza visual, apresentamos a mesma tabela com colunas alinhadas. Não adicione espaços entre as colunas na tabela de vocabulário personalizado; a tabela deve ficar desalinhada, como no exemplo anterior.

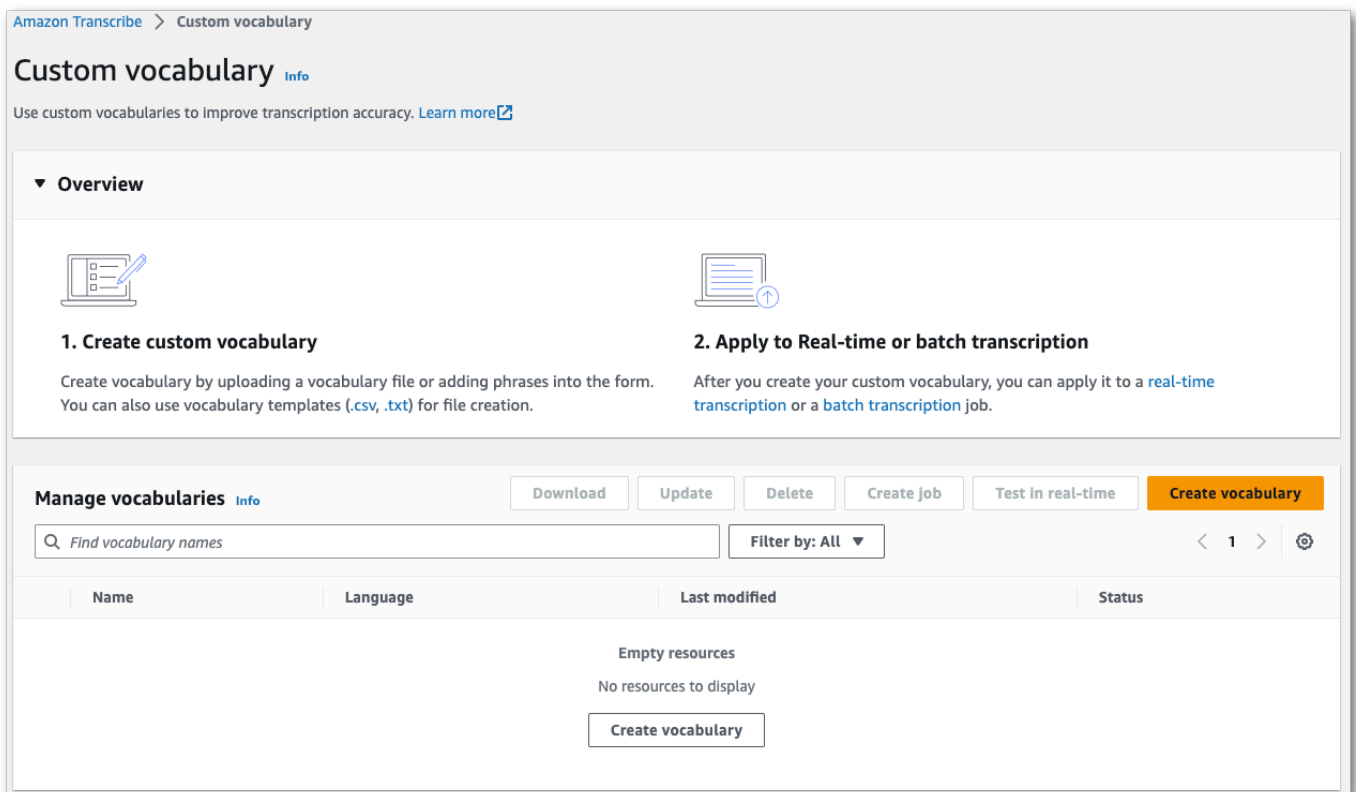
```
Phrase           [TAB]SoundsLike           [TAB]IPA           [TAB]DisplayAs
Los-Angeles     [TAB]                      [TAB]              [TAB]Los Angeles
Eva-Maria       [TAB]                      [TAB]              [TAB]
A.B.C.-s        [TAB]                      [TAB]              [TAB]ABCs
amazon-dot-com  [TAB]                      [TAB]              [TAB]amazon.com
C.L.I.          [TAB]                      [TAB]              [TAB]CLI
Andorra-la-Vella[TAB]                      [TAB]              [TAB]Andorra la Vella
Dynamo-D.B.     [TAB]                      [TAB]              [TAB]DynamoDB
V.X.-zero-two   [TAB]                      [TAB]              [TAB]VX02
V.X.-zero-two-Q.[TAB]                      [TAB]              [TAB]VX02Q
```

Criar tabelas de vocabulário personalizado

Para processar uma tabela de vocabulário personalizada para uso com Amazon Transcribe, veja os exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, selecione Vocabulário personalizado. Isso abre a página Vocabulário personalizado, na qual você pode visualizar os vocabulários existentes ou criar um.
3. Selecione Criar vocabulário.



Isso abre a página Criar vocabulário. Insira um nome para o novo vocabulário personalizado.

Aqui, você tem três opções:

- a. Carregue um arquivo txt ou csv de seu computador.

Você pode criar o vocabulário personalizado do zero ou baixar um modelo para ajudar você a começar. O vocabulário é preenchido automaticamente no painel Visualize e edite o vocabulário.

Create vocabulary [Info](#)

Vocabulary settings

Name

Vocabulary names can be up to 200 characters in length. Allowed characters: a-z, A-Z, 0-9, periods (.), dashes (-), and underscores (_).

Language

- b. Importe um arquivo txt ou csv de um Amazon S3 local.

Você pode criar o vocabulário personalizado do zero ou baixar um modelo para ajudar você a começar. Carregue o arquivo do vocabulário finalizado em um bucket do Amazon S3 e especifique o URI dele em sua solicitação. O vocabulário é preenchido automaticamente no painel Visualize e edite o vocabulário.

Create and import vocabulary [Info](#)

Vocabulary input source

File upload
Upload a vocabulary table from your computer.

S3 location
Import a vocabulary table from an S3 location.

Create vocabulary on console
Manually create a vocabulary table on the console.

Download vocabulary template – Optional
Download and complete a custom vocabulary template in your preferred format.

[Download template](#) ▼

Import from S3
Provide a path to the S3 location where your vocabulary file is stored. To find a path, go to [Amazon S3](#).

Resource URI

[View](#) [Browse S3](#)

- c. Crie manualmente o vocabulário no console.

Role até o painel Visualize e edite o vocabulário e selecione Adicionar 10 linhas. Agora você pode inserir termos manualmente.

Create and import vocabulary [Info](#)

Vocabulary input source

File upload
Upload a vocabulary table from your computer.

S3 location
Import a vocabulary table from an S3 location.

Create vocabulary on console
Manually create a vocabulary table on the console.

View and edit vocabulary (0) [Reset vocabulary](#) [Delete](#) [Download latest vocabulary](#) ▼

[Show all](#) ▼ < 1 >

Phrase ↗	SoundsLike (optional) ↗	IPA (optional) ↗	DisplayAs (optional) ↗
No rows added yet			

[Add 10 rows](#)

4. É possível editar o vocabulário no painel Visualize e edite o vocabulário. Para fazer alterações, clique na entrada que deseja modificar.

View and edit vocabulary - new (10) [Info](#) Reset vocabulary Delete Download latest ▼

Q Filter Phrase, SoundsLike, IPA or DisplayAs Show all ▼ < 1 >

<input type="checkbox"/>	Phrase ✎ ▼	SoundsLike - optional ✎ ▼	IPA - optional ✎ ▼	DisplayAs - optional ✎ ▼
<input type="checkbox"/>	Amazon-E.-C.-two	-	-	Amazon EC2
<input type="checkbox"/>	Amazon-S.-three	-	-	Amazon S3
<input type="checkbox"/>	Amazon-elasticashe	-	-	Amazon ElastiCache
<input type="checkbox"/>	Amazon-sagemaker	-	-	Amazon SageMaker
<input type="checkbox"/>	A.-W.-S.-iam	-	-	AWS IAM
<input type="checkbox"/>	A.-W.-S.-I.-o.-T.	-	-	AWS IoT
<input type="checkbox"/>	A.-W.-S.-W.-A.-F.	-	-	AWS WAF
<input type="checkbox"/>	c.-plus-plus	-	-	C++
<input type="checkbox"/>	nice-d.-c.-v.	-	-	NICE DCV
<input type="checkbox"/>	w.-w.-w.-dot-amazon-dot-com	-	-	www.amazon.com

Add row

Se cometer um erro, você receberá uma mensagem de erro detalhada para poder corrigir qualquer problema antes de processar o vocabulário. Observe que, se você não corrigir todos os erros antes de selecionar Criar vocabulário, a solicitação de vocabulário falhará.

View and edit vocabulary - new (4) [Info](#) Reset vocabulary Delete Download latest ▼

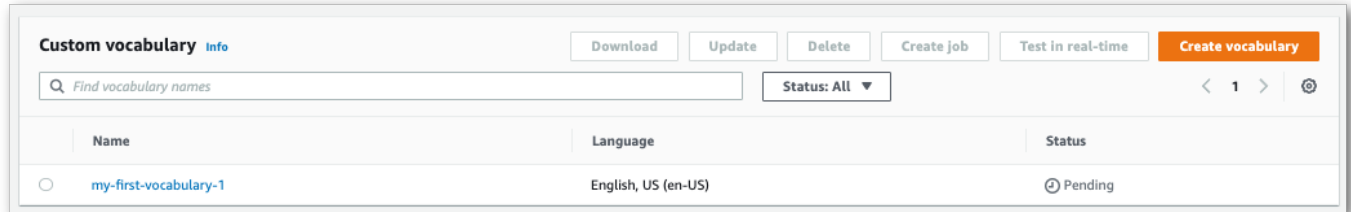
Q Filter Phrase, SoundsLike, IPA or DisplayAs Show all ▼ < 1 >

<input type="checkbox"/>	Phrase ✎ ▼	SoundsLike - optional ✎ ▼	IPA - optional ✎ ▼	DisplayAs - optional ✎ ▼
<input type="checkbox"/>	<div style="border: 1px solid red; padding: 2px;">Amazon-E.-C. two</div> ✕ ✓ ⚠ Phrase contains unsupported characters (" "). Phrase contains a formatting error.	-	-	Amazon EC2
<input type="checkbox"/>	Amazon-S.-three	-	-	Amazon S3
<input type="checkbox"/>	c.-plus-plus	-	-	C++
<input type="checkbox"/>	w.-w.-w.-dot-amazon-dot-com	-	-	www.amazon.com

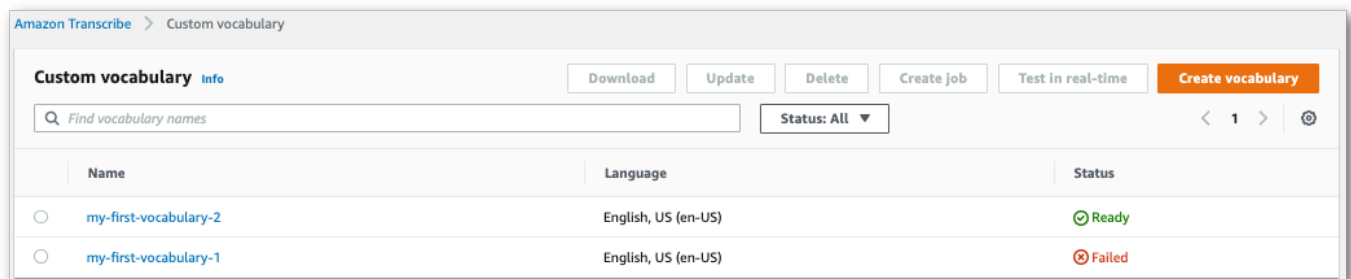
Add row

Selecione a marca de seleção (✓) para salvar as alterações ou o “X” para descartá-las.

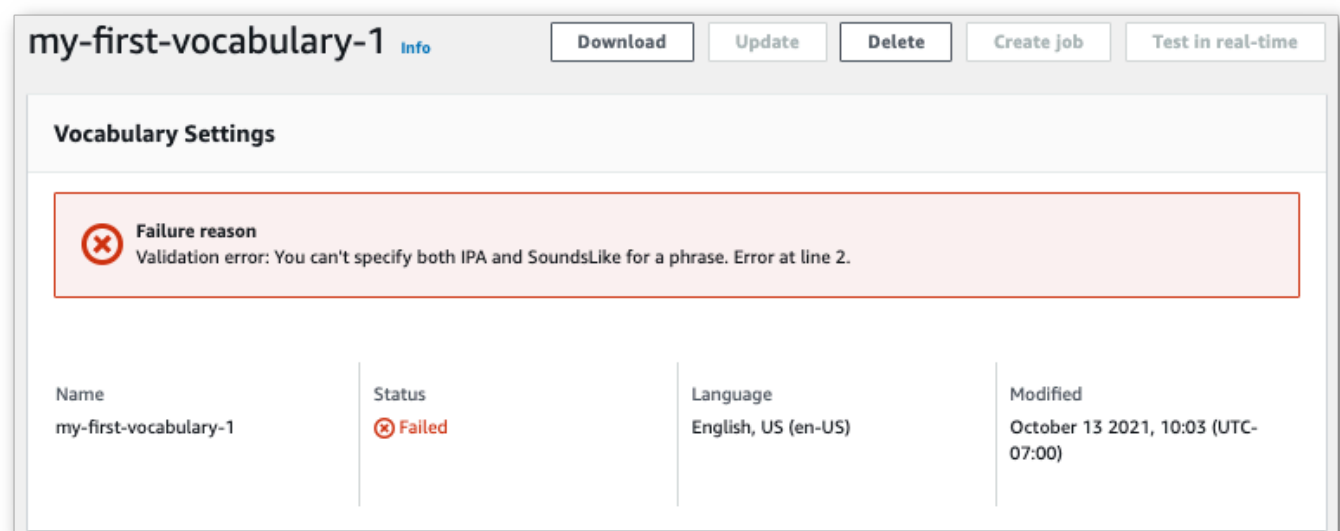
5. Se preferir, adicione tags ao vocabulário personalizado. Depois que você preencher todos os campos e estiver contente com o vocabulário, selecione Criar vocabulário na parte inferior da página. Isso leva você de volta à página Vocabulário personalizado, onde é possível ver o status do vocabulário personalizado. Quando o status muda de “Pendente” para “Pronto”, o vocabulário personalizado pode ser usado com uma transcrição.



6. Se o status mudar para “Falhou”, selecione o nome do vocabulário personalizado para acessar a página de informações.



Há um banner Motivo da falha na parte superior dessa página que fornece informações sobre o motivo da falha do vocabulário personalizado. Corrija o erro no arquivo de texto e tente novamente.



AWS CLI

Este exemplo usa o comando [create-vocabulary](#) com um arquivo de vocabulário em formato de tabela. Para ter mais informações, consulte [CreateVocabulary](#).

Para usar um vocabulário personalizado existente em um trabalho de transcrição, defina o VocabularyName no [Settings](#) campo ao chamar a [StartTranscriptionJob](#) operação ou, a partir do AWS Management Console, escolha o vocabulário personalizado na lista suspensa.

```
aws transcribe create-vocabulary \  
--vocabulary-name my-first-vocabulary \  
--vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-file.txt \  
--language-code en-US
```

Veja a seguir outro exemplo usando o comando [create-vocabulary](#) e um corpo de solicitação que cria o vocabulário personalizado.

```
aws transcribe create-vocabulary \  
--cli-input-json file://filepath/my-first-vocab-table.json
```

O arquivo *my-first-vocab-table.json* contém o seguinte corpo da solicitação.

```
{  
  "VocabularyName": "my-first-vocabulary",  
  "VocabularyFileUri": "s3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-  
table.txt",  
  "LanguageCode": "en-US"  
}
```

Assim que VocabularyState muda de PENDING para READY, seu vocabulário personalizado estará pronto para ser usado com uma transcrição. Execute o comando para visualizar o status atual do vocabulário personalizado:

```
aws transcribe get-vocabulary \  
--vocabulary-name my-first-vocabulary
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para criar um vocabulário personalizado a partir de uma tabela usando o método [create_vocabulary](#). Para ter mais informações, consulte [CreateVocabulary](#).

Para usar um vocabulário personalizado existente em um trabalho de transcrição, defina o `VocabularyName` no [Settings](#) campo ao chamar a [StartTranscriptionJob](#) operação ou, a partir do AWS Management Console, escolha o vocabulário personalizado na lista suspensa.

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary"
response = transcribe.create_vocabulary(
    LanguageCode = 'en-US',
    VocabularyName = vocab_name,
    VocabularyFileUri = 's3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-
table.txt'
)

while True:
    status = transcribe.get_vocabulary(VocabularyName = vocab_name)
    if status['VocabularyState'] in ['READY', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Note

Se você criar um novo Amazon S3 bucket para seus arquivos de vocabulário personalizados, certifique-se de que a IAM função que faz a [CreateVocabulary](#) solicitação tenha permissões para acessar esse bucket. Se o perfil não tiver as permissões corretas, sua solicitação falhará. Opcionalmente, você pode especificar uma IAM função em sua solicitação incluindo o `DataAccessRoleArn` parâmetro. Para obter mais informações sobre IAM funções e políticas em Amazon Transcribe, consulte [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#).

Criação de um vocabulário personalizado usando uma lista

Important

Vocabulários personalizados em formato de lista estão sendo descontinuados. Portanto, se você estiver criando um novo vocabulário personalizado, é altamente recomendável usar o formato de tabela.

Você pode criar vocabulários personalizados a partir de listas usando os AWS SDKs AWS Management Console AWS CLI, ou.

- AWS Management Console: Você deve criar e carregar um arquivo de texto contendo seu vocabulário personalizado. Você pode usar entradas separadas por linha ou por vírgula. Observe que sua lista deve ser salva como um arquivo de texto (*.txt) no formato. LF Se você usar qualquer outro formato, como CRLF, seu vocabulário personalizado não será aceito pelo Amazon Transcribe.
- AWS CLI e AWS SDKs: você deve incluir seu vocabulário personalizado como entradas separadas por vírgula em sua chamada de API usando a sinalização. [Phrases](#)

Se uma entrada contiver várias palavras, você deverá hifenizar cada palavra. Por exemplo, você inclui “Los Angeles” como **Los-Angeles** e “Andorra la Vella” como. **Andorra-la-Vella**

Aqui estão alguns exemplos dos dois formatos de lista válidos. Consulte [Criar listas de vocabulário personalizadas](#) para obter exemplos específicos de métodos.

- Entradas separadas por vírgula:

```
Los-Angeles,CLI,Eva-Maria,ABCs,Andorra-la-Vella
```

- Entradas separadas por linha:

```
Los-Angeles
CLI
Eva-Maria
ABCs
Andorra-la-Vella
```

⚠ Important

Você só pode usar caracteres compatíveis com o seu idioma. Consulte o [conjunto de caracteres](#) do seu idioma para obter detalhes.

As listas de vocabulário personalizadas não são suportadas com a [CreateMedicalVocabulary](#) operação. Ao criar um vocabulário médico personalizado, você deve usar um formato de tabela; consulte [Criar um vocabulário personalizado usando uma tabela](#) para obter instruções.

Criar listas de vocabulário personalizadas

Para processar uma lista de vocabulário personalizada para uso com Amazon Transcribe, veja os exemplos a seguir:

AWS CLI

Este exemplo usa o comando [create-vocabulary com um arquivo de vocabulário](#) personalizado formatado em lista. Para obter mais informações, consulte [CreateVocabulary](#).

```
aws transcribe create-vocabulary \  
--vocabulary-name my-first-vocabulary \  
--language-code en-US \  
--phrases {CLI,Eva-Maria,ABCs}
```

Aqui está outro exemplo usando o comando [create-vocabulary](#) e um corpo de solicitação que cria seu vocabulário personalizado.

```
aws transcribe create-vocabulary \  
--cli-input-json file://filepath/my-first-vocab-list.json
```

O arquivo `my-first-vocab-list.json` contém o corpo da solicitação a seguir.

```
{  
  "VocabularyName": "my-first-vocabulary",  
  "LanguageCode": "en-US",  
  "Phrases": [  
    "CLI", "Eva-Maria", "ABCs"  
  ]  
}
```

```
]
}
```

Depois de `VocabularyState` mudar de `PENDING` para `READY`, seu vocabulário personalizado estará pronto para ser usado com uma transcrição. Para visualizar o status atual do seu vocabulário personalizado, execute:

```
aws transcribe get-vocabulary \  
--vocabulary-name my-first-vocabulary
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para criar um vocabulário personalizado a partir de uma lista usando o método [create_vocabulary](#). Para obter mais informações, consulte [CreateVocabulary](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary"
response = transcribe.create_vocabulary(
    LanguageCode = 'en-US',
    VocabularyName = vocab_name,
    Phrases = [
        'CLI', 'Eva-Maria', 'ABCs'
    ]
)

while True:
    status = transcribe.get_vocabulary(VocabularyName = vocab_name)
    if status['VocabularyState'] in ['READY', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Note

Se você criar um novo Amazon S3 bucket para seus arquivos de vocabulário personalizados, certifique-se de que a IAM função que está fazendo a [CreateVocabulary](#) solicitação tenha permissões para acessar esse bucket. Se a função não tiver as permissões corretas, sua solicitação falhará. Opcionalmente, você pode especificar uma IAM função em sua solicitação incluindo o `DataAccessRoleArn` parâmetro. Para obter mais informações sobre IAM funções e políticas em Amazon Transcribe, consulte [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#).

Usar um vocabulário personalizado

Depois que seu vocabulário personalizado for criado, você poderá incluí-lo em suas solicitações de transcrição; consulte as seções a seguir para obter exemplos.

O idioma do vocabulário personalizado que você está incluindo em sua solicitação deve corresponder ao código de idioma especificado para sua mídia. Se os idiomas não corresponderem, seu vocabulário personalizado não será aplicado à sua transcrição e não haverá avisos ou erros.

Usando um vocabulário personalizado em uma transcrição em lote

Para usar um vocabulário personalizado com uma transcrição em lote, veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Trabalhos de transcrição e selecione Criar tarefa (canto superior direito). Isso abre a página Especificar detalhes do trabalho.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

[▶ Additional settings](#)

Dê um nome ao seu trabalho e especifique sua mídia de entrada. Opcionalmente, inclua outros campos e escolha Avançar.

3. Na parte inferior da página Configurar tarefa, no painel Personalização, ative Vocabulário personalizado.

Configure job - *optional* [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Vocabulary selection
The vocabularies shown here are based on your language settings. You can choose up to one vocabulary per language. You can also [create a new vocabulary](#). [↗](#)

Choose a vocabulary ▼

Cancel Previous Create job

4. Selecione seu vocabulário personalizado no menu suspenso.

Selecione Criar tarefa para executar sua tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o `Settings` parâmetro com o `VocabularyName` subparâmetro. Para obter mais informações, consulte [StartTranscriptionJob](#) e [Settings](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--settings VocabularyName=my-first-vocabulary
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que inclui seu vocabulário personalizado com esse trabalho.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-vocabulary-job.json
```

O arquivo `my-first-vocabulary-job.json` contém o seguinte corpo da solicitação.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Settings": {  
    "VocabularyName": "my-first-vocabulary"  
  }  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para incluir um vocabulário personalizado usando o `Settings` argumento para o método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#) e [Settings](#).

Para obter exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e serviços cruzados, consulte o [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#) capítulo.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Settings = {
        'VocabularyName': 'my-first-vocabulary'
    }
)

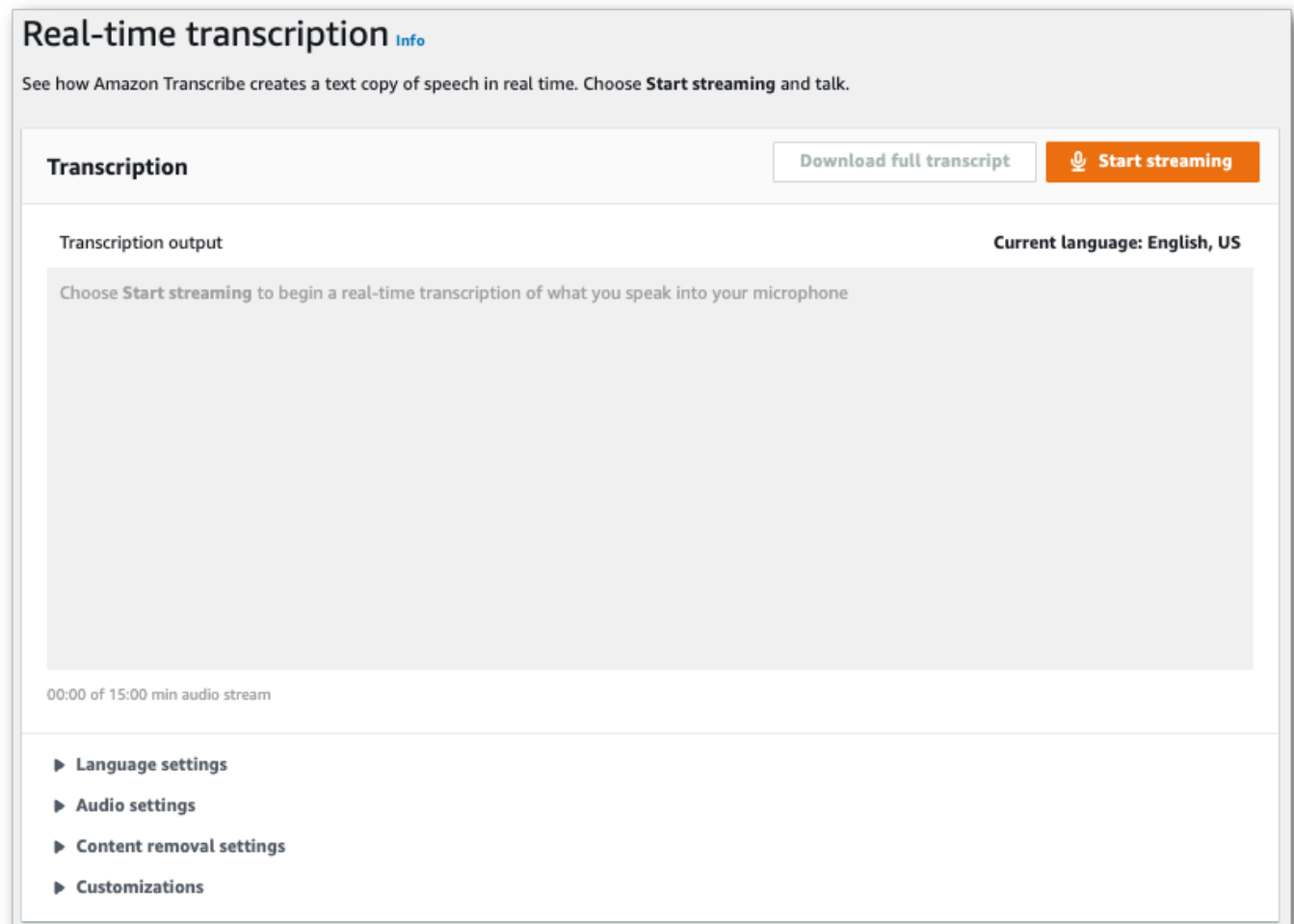
while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Usando um vocabulário personalizado em uma transcrição de streaming

Para usar um vocabulário personalizado com uma transcrição em streaming, veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Transcrição em tempo real. Role para baixo até Personalizações e expanda esse campo se ele estiver minimizado.



Real-time transcription [Info](#)

See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.

Transcription Download full transcript Start streaming

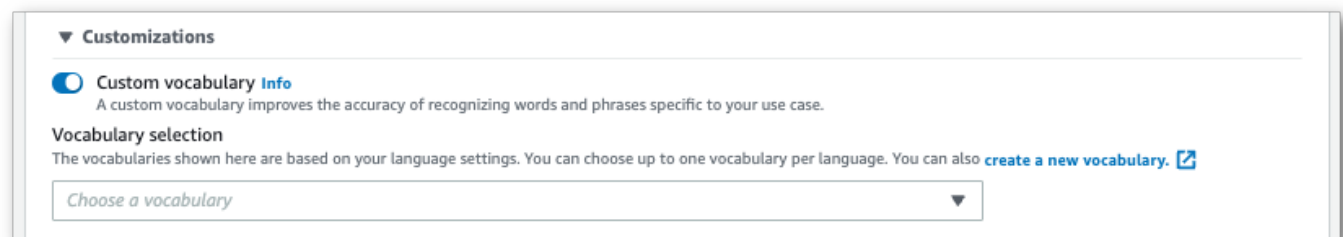
Transcription output Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone

00:00 of 15:00 min audio stream

- ▶ Language settings
- ▶ Audio settings
- ▶ Content removal settings
- ▶ Customizations

3. Ative Vocabulário personalizado e selecione um vocabulário personalizado no menu suspenso.



▼ Customizations

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Vocabulary selection
The vocabularies shown here are based on your language settings. You can choose up to one vocabulary per language. You can also [create a new vocabulary](#).

Choose a vocabulary ▼

Inclua outras configurações que deseja aplicar ao seu stream.

4. Agora está tudo pronto para transcrição do seu stream. Selecione Iniciar transmissão e começar a falar. Para finalizar seu ditado, selecione Interromper streaming.

Transcrição do HTTP/2

Este exemplo cria uma solicitação HTTP/2 que inclui seu vocabulário personalizado. Para obter mais informações sobre como usar o streaming HTTP/2 com Amazon Transcribe, consulte [Configurar um fluxo de HTTP/2](#). Para obter mais detalhes sobre parâmetros e cabeçalhos específicos para Amazon Transcribe, consulte [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-vocabulary-name: my-first-vocabulary
transfer-encoding: chunked
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

WebSocket riacho

Este exemplo cria um URL predefinido que aplica seu vocabulário personalizado a um WebSocket stream. Quebras de linha foram adicionadas para uma melhor leitura. Para obter mais informações sobre como usar WebSocket streams com Amazon Transcribe, consulte [Configurando um WebSocket stream](#). Para obter mais detalhes sobre os parâmetros, consulte [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
```

```
&language-code=en-US  
&media-encoding=flac  
&sample-rate=16000  
&vocabulary-name=my-first-vocabulary
```


As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Modelos de linguagem personalizados

Modelos de linguagem personalizados são projetados para melhorar a precisão da transcrição para fala de um domínio específico. Isso inclui qualquer conteúdo fora do que você ouviria em conversas normais do dia a dia. Por exemplo, se você estiver transcrevendo os anais de uma conferência científica, é improvável que uma transcrição padrão reconheça muitos dos termos científicos usados pelos apresentadores. Nesse caso, você pode treinar um modelo de linguagem personalizado para reconhecer os termos especializados usados em sua disciplina.

Diferentemente dos vocabulários personalizados, que aumentam o reconhecimento de uma palavra fornecendo dicas (como pronúncias), os modelos de linguagem personalizados aprendem o contexto associado a uma determinada palavra. Isso inclui como e quando uma palavra é usada e a relação que uma palavra tem com outras palavras. Por exemplo, se você treinar seu modelo usando trabalhos de pesquisa em ciências climáticas, seu modelo poderá aprender que “bloco de gelo” é um par de palavras mais provável do que “fluxo de gelo”.

Para ver os idiomas suportados para modelos de linguagem personalizados, consulte [idiomas oferecidos e recursos específicos do idioma](#). Observe que, se você incluir um modelo de idioma personalizado em sua solicitação, não poderá ativar a identificação do idioma (você deve especificar um código de idioma).

 Operações de API específicas para modelos de linguagem personalizados

[CreateLanguageModel](#), [DeleteLanguageModel](#), [DescribeLanguageModel](#),
[ListLanguageModels](#)

Fontes de dados

Você pode usar qualquer tipo de dados de texto que quiser para treinar seu modelo. No entanto, quanto mais próximo o conteúdo do texto estiver do conteúdo de áudio, mais preciso será o modelo.

Portanto, é importante escolher dados de texto que usem os mesmos termos no mesmo contexto do áudio.

Os melhores dados para treinar um modelo são transcrições precisas. Isso é considerado dado no domínio. Os dados de texto no domínio têm exatamente os mesmos termos, uso e contexto do áudio que você deseja transcrever.

Se você não tiver transcrições precisas, use artigos de periódicos, relatórios técnicos, documentos técnicos, anais de conferências, manuais de instruções, artigos de notícias, conteúdo do site e qualquer outro texto que contenha os termos desejados usados em um contexto semelhante ao do seu áudio. Isso é considerado dado relacionado ao domínio.

A criação de um modelo de idioma personalizado robusto pode exigir uma quantidade significativa de dados de texto, que devem conter os termos falados em seu áudio. Você pode Amazon Transcribe fornecer até 2 GB de dados de texto para treinar seu modelo — isso é chamado de dados de treinamento. Opcionalmente, quando você não tem (ou poucas) transcrições no domínio, você pode Amazon Transcribe fornecer até 200 MB de dados de texto para ajustar seu modelo — isso é chamado de ajuste de dados.

Treinamento versus ajuste de dados

O objetivo dos dados de treinamento é ensinar Amazon Transcribe a reconhecer novos termos e aprender o contexto no qual esses termos são usados. Para criar um modelo robusto, Amazon Transcribe pode ser necessário um grande volume de dados de texto relevantes. É altamente recomendável fornecer o máximo possível de dados de treinamento, até o limite de 2 GB.

O objetivo de ajustar os dados é ajudar a refinar e otimizar as relações contextuais aprendidas com seus dados de treinamento. Dados de ajuste não são necessários para criar um modelo de idioma personalizado.

Cabe a você decidir a melhor forma de selecionar o treinamento e, opcionalmente, ajustar os dados. Cada caso é único e depende do tipo e da quantidade de dados que você tem. Os dados de ajuste são recomendados quando você não tem dados de treinamento no domínio.

Se você optar por incluir os dois tipos de dados, não sobreponha seus dados de treinamento e ajuste; os dados de treinamento e ajuste devem ser exclusivos. A sobreposição de dados pode influenciar e distorcer seu modelo de linguagem personalizado, afetando sua precisão.

Como orientação geral, recomendamos usar texto preciso no domínio como dados de treinamento sempre que possível. Aqui estão alguns cenários gerais, listados em ordem de preferência:

- Se você tiver mais de 10.000 palavras de texto transcrito preciso no domínio, use-o como dados de treinamento. Nesse caso, não há necessidade de incluir dados de ajuste. Esse é o cenário ideal para treinar um modelo de idioma personalizado.
- Se você tem um texto de transcrição preciso no domínio que contém menos de 10.000 palavras e não está obtendo os resultados desejados, considere aumentar seus dados de treinamento com textos escritos relacionados ao domínio, como relatórios técnicos. Nesse caso, reserve uma pequena parte (10 a 25%) dos dados de transcrição no domínio para usar como dados de ajuste.
- Se você não tiver nenhum texto de transcrição no domínio, faça upload de todo o texto relacionado ao domínio como dados de treinamento. Nesse caso, o texto no estilo transcrição é preferível ao texto escrito. Esse é o cenário menos eficaz para treinar um modelo de linguagem personalizado.

Quando estiver pronto para criar seu modelo, consulte [Criação de um modelo de linguagem personalizado](#).

Criação de um modelo de linguagem personalizado

Antes de criar seu modelo de idioma personalizado, você deve:

- Preparar seus dados. Os dados devem ser salvos em formato de texto simples e não podem conter caracteres especiais.
- Carregue seus dados em um Amazon S3 bucket. É recomendável criar pastas separadas para treinamento e ajuste de dados.
- Verifique se Amazon Transcribe tem acesso ao seu Amazon S3 bucket. Você deve especificar uma IAM função que tenha permissões de acesso para usar seus dados.

Preparação de seus dados

Você pode compilar todos os seus dados em um arquivo ou salvá-los como vários arquivos. Observe que, se você optar por incluir dados de ajuste, eles deverão ser salvos em um arquivo separado dos dados de treinamento.

Não importa quantos arquivos de texto você usa para seus dados de treinamento ou ajuste. O upload de um arquivo com 100.000 palavras produz o mesmo resultado do upload de 10 arquivos com 10.000 palavras. Prepare seus dados de texto da maneira mais conveniente para você.

Certifique-se de que todos os seus arquivos de dados atendam aos seguintes critérios:

- Eles estão todos na mesma linguagem do modelo que você deseja criar. Por exemplo, se você quiser criar um modelo de idioma personalizado que transcreva áudio em inglês dos EUA (en-US), todos os seus dados de texto devem estar em inglês dos EUA.
- Eles estão no formato de texto sem formatação com codificação UTF-8.
- Eles não contêm nenhum caractere ou formatação especial, como tags HTML.
- Eles totalizam um total máximo combinado de 2 GB para dados de treinamento e 200 MB para dados de ajuste.

Se algum desses critérios não for atendido, o modelo falhará.

Upload de seus dados

Antes de carregar seus dados, crie uma nova pasta para seus dados de treinamento. Se estiver usando dados de ajuste, crie outra pasta separada.

Os URIs de seus buckets podem ter a seguinte aparência:

- `s3://DOC-EXAMPLE-BUCKET/my-model-training-data/`
- `s3://DOC-EXAMPLE-BUCKET/my-model-tuning-data/`

Carregue seus dados de treinamento e ajuste nos compartimentos apropriados.

Você pode adicionar mais dados a esses compartimentos em uma data posterior. No entanto, se você fizer isso, precisará recriar seu modelo com os novos dados. Os modelos existentes não podem ser atualizados com novos dados.

Permitindo o acesso aos seus dados

Para criar um modelo de linguagem personalizado, você deve especificar uma IAM função que tenha permissões para acessar seu Amazon S3 bucket. Se você ainda não tem uma função com acesso ao Amazon S3 bucket em que colocou seus dados de treinamento, você deve criar uma. Depois de criar uma função, é possível anexar uma política para conceder permissões a essa função. Não anexar uma política a um usuário.

Para obter exemplos de políticas, consulte [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#).

Para saber como criar uma nova IAM identidade, consulte [IAM identidades \(usuários, grupos de usuários e funções\)](#).

Para saber mais sobre as políticas, consulte:

- [Políticas e permissões em IAM](#)
- [Criação de IAM políticas](#)
- [Gerenciamento de acesso para recursos da AWS](#)

Criando seu modelo de linguagem personalizado

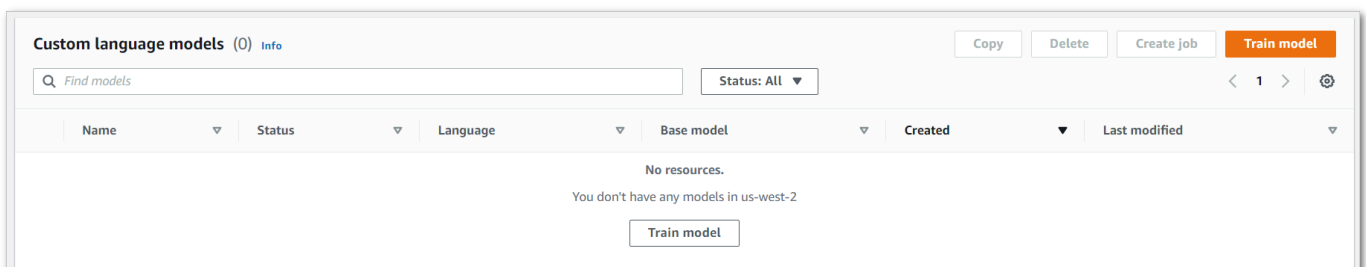
Ao criar seu modelo de linguagem personalizado, você deve escolher um modelo base. Há duas opções de modelo básico:

- **NarrowBand:** Use essa opção para áudio com uma taxa de amostragem inferior a 16.000 Hz. Esse tipo de modelo é normalmente usado para conversas telefônicas gravadas a 8.000 Hz.
- **WideBand:** Use essa opção para áudio com uma taxa de amostragem maior ou igual a 16.000 Hz.

Você pode criar modelos de linguagem personalizados usando o AWS Management Console, AWS CLI, ou AWS SDKs.; veja os exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Modelo de idioma personalizado. Isso abre a página Modelos de idioma personalizados, na qual você pode visualizar os modelos de linguagem personalizados existentes ou treinar um novo modelo de idioma personalizado.
3. Para treinar um novo modelo, selecione Treinar modelo.



Isso leva você à página do modelo do Train. Adicione um nome, especifique o idioma e escolha o modelo básico que você deseja para seu modelo. Em seguida, adicione o caminho ao seu treinamento e, opcionalmente, seus dados de ajuste. Você deve incluir uma IAM função que tenha permissões para acessar seus dados.

Train model [Info](#)

Model settings

Name

The name can be up to 200 characters long. Valid characters: A-Z, a-z, 0-9, and _ - (hyphen).

Language

Choose the language of your model.

 ▼

Base model [Info](#)

Choose the base model that you want to use to create your custom language model. Choose the model based on the sample rate of your source audio.

Narrow band

For audio that has a sample rate less than 16 KHz. Typically, this is 8 KHz audio from telephone conversations.

Wide band

For audio that has a sample rate of 16 KHz or greater. Typically, this is 16 KHz audio from media sources.

Training data [Info](#)

Training data location on S3

Type or paste the S3 prefix for the text files that you want to use as training data, or browse to find the files that have matching S3 prefixes.

The file format must be plain text in the language that you have selected for the model. The maximum file size is 2 GB.

Tuning data - optional [Info](#)

Tuning data location on S3

Type or paste the S3 prefix for the text files that you want to use as tuning data, or browse to find the files that have matching S3 prefixes.

The file format must be plain text in the language that you have selected for the model. The maximum file size is 200 MB.

Access permissions

IAM role [Info](#)

Use an existing IAM role

Create an IAM role

By choosing **Train model** you are authorizing creation of this role.

Role name

A role that grants access to the S3 input locations.

 ▼

4. Depois de preencher todos os campos, selecione Modelo de trem na parte inferior da página.

AWS CLI

Este exemplo usa o [create-language-model](#) comando. Para ter mais informações, consulte [CreateLanguageModel](#) e [LanguageModel](#).

```
aws transcribe create-language-model \  
--base-model-name NarrowBand \  
--model-name my-first-language-model \  
--input-data-config S3Uri=s3://DOC-EXAMPLE-BUCKET/my-clm-training-  
data/,TuningDataS3Uri=s3://DOC-EXAMPLE-BUCKET/my-clm-tuning-  
data/,DataAccessRoleArn=arn:aws:iam::111122223333:role/ExampleRole \  
--language-code en-US
```

Aqui está outro exemplo usando o [create-language-model](#) comando e um corpo de solicitação que cria seu modelo de linguagem personalizado.

```
aws transcribe create-language-model \  
--cli-input-json file://filepath/my-first-language-model.json
```

O arquivo `my-first-language-model.json` contém o seguinte corpo da solicitação.

```
{  
  "BaseModelName": "NarrowBand",  
  "ModelName": "my-first-language-model",  
  "InputDataConfig": {  
    "S3Uri": "s3://DOC-EXAMPLE-BUCKET/my-clm-training-data/",  
    "TuningDataS3Uri": "s3://DOC-EXAMPLE-BUCKET/my-clm-tuning-data/",  
    "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole"  
  },  
  "LanguageCode": "en-US"  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para criar um CLM usando o método [create_language_model](#). Para ter mais informações, consulte [CreateLanguageModel](#) e [LanguageModel](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
model_name = 'my-first-language-model',
transcribe.create_language_model(
    LanguageCode = 'en-US',
    BaseModelName = 'NarrowBand',
    ModelName = model_name,
    InputDataConfig = {
        'S3Uri': 's3://DOC-EXAMPLE-BUCKET/my-clm-training-data/',
        'TuningDataS3Uri': 's3://DOC-EXAMPLE-BUCKET/my-clm-tuning-data/',
        'DataAccessRoleArn': 'arn:aws:iam::111122223333:role/ExampleRole'
    }
)

while True:
    status = transcribe.get_language_model(ModelName = model_name)
    if status['LanguageModel']['ModelStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Atualizando seu modelo de idioma personalizado

Amazon Transcribe atualiza continuamente os modelos básicos disponíveis para modelos de linguagem personalizados. Para se beneficiar dessas atualizações, recomendamos treinar novos modelos de linguagem personalizados a cada 6 a 12 meses.

Para ver se seu modelo de linguagem personalizado está usando o modelo base mais recente, execute uma [DescribeLanguageModel](#) solicitação usando o AWS CLI ou um AWS SDK e encontre o `UpgradeAvailability` campo em sua resposta.

Se `UpgradeAvailability` estiver `true`, seu modelo não está executando a versão mais recente do modelo básico. Para usar o modelo base mais recente em um modelo de idioma personalizado, você deve criar um novo modelo de idioma personalizado. Modelos de linguagem personalizados não podem ser atualizados.

Usando um modelo de linguagem personalizado

Depois de criar seu modelo de linguagem personalizado, você pode incluí-lo em suas solicitações de transcrição; consulte as seções a seguir para obter exemplos.

O idioma do modelo que você está incluindo em sua solicitação deve corresponder ao código de idioma especificado para sua mídia. Se os idiomas não corresponderem, seu modelo de idioma personalizado não será aplicado à sua transcrição e não haverá avisos ou erros.

Usando um modelo de linguagem personalizado em uma transcrição em lote

Para usar um modelo de linguagem personalizado com uma transcrição em lote, veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Trabalhos de transcrição e selecione Criar tarefa (canto superior direito). Isso abre a página Especificar detalhes do trabalho.
3. No painel Configurações do Job, em Tipo de modelo, selecione a caixa Modelo de idioma personalizado.

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language

Choose the language of the input audio.

Custom model selection

Choose an existing model or [create a new one.](#)

► **Additional settings**

Você também deve selecionar um idioma de entrada no menu suspenso.

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language

Choose the language of the input audio.

English, US (en-US) ▲

English, US (en-US)

English, AU (en-AU)

English, UK (en-GB)

Hindi, IN (hi-IN)

Spanish, US (es-US)

- Em Seleção de modelo personalizado, selecione um modelo de idioma personalizado existente no menu suspenso ou Crie um novo.

Adicione a Amazon S3 localização do seu arquivo de entrada no painel Dados de entrada.

- Selecione Avançar para obter mais opções de configuração.

Selecione Criar tarefa para executar sua tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o `ModelSettings` parâmetro com o `VocabularyName` subparâmetro. Para obter mais informações, consulte [StartTranscriptionJob](#) e [ModelSettings](#).

```
aws transcribe start-transcription-job \
```

```
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--model-settings LanguageModelName=my-first-language-model
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que inclui seu modelo de linguagem personalizado com essa tarefa.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-model-job.json
```

O arquivo `my-first-model-job.json` contém o seguinte corpo da solicitação.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "ModelSettings": {  
    "LanguageModelName": "my-first-language-model"  
  }  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para incluir um modelo de linguagem personalizado usando o `ModelSettings` argumento para o método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#) e [ModelSettings](#).

Para obter exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e serviços cruzados, consulte o [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#) capítulo.

```
from __future__ import print_function
```

```
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    ModelSettings = {
        'LanguageModelName': 'my-first-language-model'
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Usando um modelo de linguagem personalizado em uma transcrição de streaming


Para usar um modelo de linguagem personalizado com uma transcrição de streaming, veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Transcrição em tempo real. Role para baixo até Personalizações e expanda esse campo se ele estiver minimizado.

Real-time transcription [Info](#)

See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.

Transcription Download full transcript  **Start streaming**

Transcription output Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone

00:00 of 15:00 min audio stream

- ▶ **Language settings**
- ▶ **Audio settings**
- ▶ **Content removal settings**
- ▶ **Customizations**


3. Ative o modelo de idioma personalizado e selecione um modelo no menu suspenso.

▼ **Customizations**

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Partial results stabilization [Info](#)
Configure Amazon Transcribe to present results that don't change as it processes the transcription output from your stream.

Custom language model [Info](#)
Select the model you want to use with this streaming session.

Custom model selection [Info](#)
The models shown here are based on your language settings. You can choose up to one model per language. You can also [create a new model](#). 

Choose model ▼

Inclua as outras configurações que deseja aplicar à sua transmissão.

4. Agora está tudo pronto para transmitir sua transmissão. Selecione Iniciar transmissão e começar a falar. Para finalizar seu ditado, selecione Interromper streaming.

fluxo HTTP/2 do HTTP/

Este exemplo cria uma solicitação HTTP/2 que inclui seu modelo de linguagem personalizado. Para obter mais informações sobre como usar o streaming HTTP/2 com Amazon Transcribe, consulte [Configurar um fluxo de HTTP/2](#). Para obter mais detalhes sobre parâmetros e cabeçalhos específicos para Amazon Transcribe, consulte [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-language-model-name: my-first-language-model
transfer-encoding: chunked
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

WebSocket stream

Este exemplo cria um URL pré-assinado que aplica seu modelo de linguagem personalizado a um WebSocket stream. Quebras de linha foram adicionadas para uma melhor leitura. Para obter mais informações sobre como usar WebSocket streams com Amazon Transcribe, consulte [Configurando um WebSocket stream](#). Para obter mais detalhes sobre os parâmetros, consulte [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
```

```
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date  
&language-code=en-US  
&media-encoding=flac  
&sample-rate=16000  
&language-model-name=my-first-language-model
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Usando filtros de vocabulário personalizados para excluir, mascarar ou sinalizar palavras

Um filtro de vocabulário personalizado é um arquivo de texto que contém uma lista personalizada de palavras individuais que você deseja modificar na saída da transcrição.

Um caso de uso comum é a remoção de termos ofensivos ou profanos; no entanto, os filtros de vocabulário personalizados são totalmente personalizados, para que você possa selecionar as palavras que quiser. Por exemplo, se você tiver um novo produto prestes a ser lançado, poderá mascarar o nome do produto nas transcrições da reunião. Nesse caso, você mantém as partes interessadas up-to-date enquanto mantém o nome do produto em segredo até o lançamento.

A filtragem de vocabulário tem três métodos de exibição: `mask`, `remove`, `tag` e. Consulte os exemplos a seguir para ver como cada um funciona.

- **Máscara:** substitui palavras especificadas por três asteriscos (`***`).

```
"transcript": "You can specify a list of *** or *** words, and *** *** removes them from transcripts automatically."
```

- **Remove:** exclui palavras especificadas, não deixando nada em seu lugar.

```
"transcript": "You can specify a list of or words, and removes them from transcripts automatically."
```

- **Tag:** adiciona uma tag (`"vocabularyFilterMatch": true`) a cada palavra especificada, mas não altera a palavra em si. A marcação permite substituições e edições rápidas de transcrições.

```
"transcript": "You can specify a list of profane or offensive words, and amazon transcribe removes them from transcripts automatically."
```

```
...
```

```
  "alternatives": [  
    {  
      "confidence": "1.0",  
      "content": "profane"  
    }  
  ],  
  "type": "pronunciation",  
  "vocabularyFilterMatch": true
```

Ao enviar uma solicitação de transcrição, você pode especificar um filtro de vocabulário personalizado e o método de filtragem que deseja aplicar. Amazon Transcribe em seguida, modifica as correspondências exatas de palavras quando elas aparecem na sua transcrição, de acordo com o método de filtragem que você especificar.

Filtros de vocabulário personalizados podem ser aplicados a solicitações de transcrição em lote e streaming. Para saber como criar um filtro de banco de dados personalizado, consulte [Criação de um filtro de vocabulário](#). Para saber como aplicar seu filtro de vocabulário personalizado, consulte [Usando um filtro de vocabulário personalizado](#).

Note

Amazon Transcribe mascara automaticamente termos racialmente sensíveis, embora você possa desativar esse filtro padrão entrando em contato com o [Support AWS Técnico](#).

Para ver um passo a passo em vídeo sobre a filtragem de vocabulário, consulte [Usando filtros de vocabulário](#).

Operações de API específicas para filtragem de vocabulário

[CreateVocabularyFilter](#), [DeleteVocabularyFilter](#), [GetVocabularyFilter](#), [ListVocabularyFilters](#), [UpdateVocabularyFilter](#)

Criação de um filtro de vocabulário

Há duas opções para criar um filtro de vocabulário personalizado:

1. Salve uma lista de palavras separadas por linha como um arquivo de texto simples com codificação UTF-8.
 - Você pode usar essa abordagem com os [AWS SDKs](#), [AWS Management Console](#), [AWS CLI](#), ou.
 - Se estiver usando o [AWS Management Console](#), você pode fornecer um caminho local ou um [Amazon S3 URI](#) para seu arquivo de vocabulário personalizado.
 - Se estiver usando os [AWS SDKs](#), [AWS CLI](#) ou, você deve carregar seu arquivo de vocabulário personalizado em um [Amazon S3 bucket](#) e incluir o [Amazon S3 URI](#) em sua solicitação.
2. Inclua uma lista de palavras separadas por vírgula diretamente em sua solicitação de API.

- Você pode usar essa abordagem com os [AWS SDKs](#) [AWS CLI](#) ou usando o [Words](#) parâmetro.

Para obter exemplos de cada método, consulte [Criação de filtros de vocabulário personalizados](#)

Coisas a serem observadas ao criar seu filtro de vocabulário personalizado:

- As palavras não diferenciam maiúsculas de minúsculas. Por exemplo, “maldição” e “MALDIÇÃO” são tratadas da mesma forma.
- Somente correspondências exatas de palavras são filtradas. Por exemplo, se seu filtro incluir “xingar”, mas sua mídia contiver a palavra “xingar” ou “xingar”, eles não serão filtrados. Somente casos de “palavrão” são filtrados. Portanto, você deve incluir todas as variações das palavras que deseja filtrar.
- Os filtros não se aplicam a palavras contidas em outras palavras. Por exemplo, se um filtro de vocabulário personalizado contiver “marinho”, mas não “submarino”, “submarino” não será alterado na transcrição.
- Cada entrada só pode conter uma palavra (sem espaços).
- Se você salvar seu filtro de vocabulário personalizado como um arquivo de texto, ele deverá estar em formato de texto simples com codificação UTF-8.
- Você pode ter até 100 filtros de vocabulário personalizados por cada uma [Conta da AWS](#) e cada um pode ter até 50 Kb de tamanho.
- Você só pode usar caracteres compatíveis com seu idioma. Consulte o [conjunto de caracteres](#) do seu idioma para obter detalhes.

Criação de filtros de vocabulário personalizados

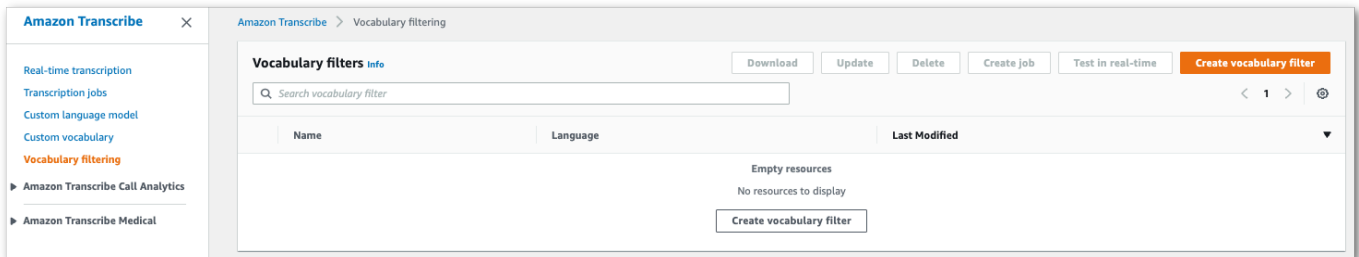
Para processar um filtro de vocabulário personalizado para uso com [Amazon Transcribe](#), veja os exemplos a seguir:

AWS Management Console

Antes de continuar, salve seu filtro de vocabulário personalizado como um arquivo de texto (*.txt). Opcionalmente, você pode carregar seu arquivo em um [Amazon S3 bucket](#).

1. Faça login no [AWS Management Console](#).

2. No painel de navegação, escolha Filtragem de vocabulário. Isso abre a página Filtros de vocabulário, na qual você pode ver os filtros de vocabulário personalizados existentes ou criar um novo.
3. Selecione Criar filtro de vocabulário.



Você irá para a página do filtro de Criação de vocabulário. Insira um nome para o seu novo filtro de vocabulário personalizado.

Selecione a opção Upload de arquivo ou localização do S3 em Fonte de entrada de vocabulário. Em seguida, especifique a localização do seu arquivo de vocabulário personalizado.

Create vocabulary filter Info

Vocabulary filtering settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9 and - (hyphen).

Language

Vocabulary input source Info

File upload

S3 location

Vocabulary filter file location on S3

Provide a path to the S3 location where your vocabulary filter file is stored. To find a path, go to [Amazon S3](#)

File format: txt, maximum size 50 KB.

Tags - optional

A tag is a label you can add to a resource as metadata to help you organize, search, or filter your data. Each tag consists of a key and an optional value, in the form 'key:value'.

No tags associated with the resource.

You can add up to 50 more tags.

4. Opcionalmente, adicione tags ao seu filtro de vocabulário personalizado. Depois de preencher todos os campos, selecione Criar filtro de vocabulário na parte inferior da página. Se não houver erros ao processar seu arquivo, isso o levará de volta à página de filtros de vocabulário.

Seu filtro de vocabulário personalizado agora está pronto para uso.

AWS CLI

Este exemplo usa o [create-vocabulary-filter](#) comando para processar uma lista de palavras em um filtro de vocabulário personalizado utilizável. Para obter mais informações, consulte [CreateVocabularyFilter](#).

Opção 1: você pode incluir sua lista de palavras em sua solicitação usando o `owords` parâmetro.


```
aws transcribe create-vocabulary-filter \
--vocabulary-filter-name my-first-vocabulary-filter \
--language-code en-US \
--words profane,offensive,Amazon,Transcribe
```

Opção 2: você pode salvar sua lista de palavras como um arquivo de texto e carregá-la em um Amazon S3 bucket e, em seguida, incluir o URI do arquivo em sua solicitação usando o `vocabulary-filter-file-uri` parâmetro.

```
aws transcribe create-vocabulary-filter \
--vocabulary-filter-name my-first-vocabulary-filter \
--language-code en-US \
--vocabulary-filter-file-uri s3://DOC-EXAMPLE-BUCKET/my-vocabulary-filters/my-vocabulary-filter.txt
```

Aqui está outro exemplo usando o [create-vocabulary-filter](#) comando e um corpo de solicitação que cria seu filtro de vocabulário personalizado.

```
aws transcribe create-vocabulary-filter \
--cli-input-json file://filepath/my-first-vocab-filter.json
```

O arquivo `my-first-vocab-filter.json` contém o seguinte corpo da solicitação.

Opção 1: você pode incluir sua lista de palavras em sua solicitação usando o `Words` parâmetro.

```
{
  "VocabularyFilterName": "my-first-vocabulary-filter",
  "LanguageCode": "en-US",
  "Words": [
    "profane", "offensive", "Amazon", "Transcribe"
  ]
}
```

Opção 2: você pode salvar sua lista de palavras como um arquivo de texto e carregá-la em um Amazon S3 bucket e, em seguida, incluir o URI do arquivo em sua solicitação usando o `VocabularyFilterFileUri` parâmetro.

```
{
  "VocabularyFilterName": "my-first-vocabulary-filter",
  "LanguageCode": "en-US",
```

```
"VocabularyFilterFileUri": "s3://DOC-EXAMPLE-BUCKET/my-vocabulary-filters/my-  
vocabulary-filter.txt"  
}
```

Note

Se você incluir `VocabularyFilterFileUri` em sua solicitação, não poderá usar `Words`; você deve escolher um ou outro.

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para criar um filtro de vocabulário personalizado usando o método [create_vocabulary_filter](#). Para obter mais informações, consulte [CreateVocabularyFilter](#).

Para obter exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e serviços cruzados, consulte o [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#) capítulo.

Opção 1: você pode incluir sua lista de palavras em sua solicitação usando o `Words` parâmetro.

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
vocab_name = "my-first-vocabulary-filter"  
response = transcribe.create_vocabulary_filter(  
    LanguageCode = 'en-US',  
    VocabularyFilterName = vocab_name,  
    Words = [  
        'profane', 'offensive', 'Amazon', 'Transcribe'  
    ]  
)
```

Opção 2: você pode salvar sua lista de palavras como um arquivo de texto e carregá-la em um Amazon S3 bucket e, em seguida, incluir o URI do arquivo em sua solicitação usando o `VocabularyFilterFileUri` parâmetro.

```
from __future__ import print_function
```

```
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary-filter"
response = transcribe.create_vocabulary_filter(
    LanguageCode = 'en-US',
    VocabularyFilterName = vocab_name,
    VocabularyFilterFileUri = 's3://DOC-EXAMPLE-BUCKET/my-vocabulary-filters/my-
vocabulary-filter.txt'
)
```

Note

Se você incluir `VocabularyFilterFileUri` em sua solicitação, não poderá usar `Words`; você deve escolher um ou outro.

Note

Se você criar um novo Amazon S3 intervalo para seus arquivos de filtro de vocabulário personalizados, certifique-se de que a IAM função que está fazendo a [CreateVocabularyFilter](#) solicitação tenha permissões para acessar esse intervalo. Se a função não tiver as permissões corretas, sua solicitação falhará. Opcionalmente, você pode especificar uma IAM função em sua solicitação incluindo o `DataAccessRoleArn` parâmetro. Para obter mais informações sobre IAM funções e políticas em Amazon Transcribe, consulte [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#).

Usando um filtro de vocabulário personalizado

Depois que seu filtro de vocabulário personalizado for criado, você poderá incluí-lo em suas solicitações de transcrição; consulte as seções a seguir para obter exemplos.

O idioma do filtro de vocabulário personalizado que você está incluindo em sua solicitação deve corresponder ao código de idioma especificado para sua mídia. Se você usar a identificação de idioma e especificar várias opções de idioma, poderá incluir um filtro de vocabulário personalizado por idioma especificado. Se os idiomas dos seus filtros de vocabulário personalizados não corresponderem ao idioma identificado em seu áudio, seus filtros não serão aplicados à sua transcrição e não haverá avisos ou erros.

Usando um filtro de vocabulário personalizado em uma transcrição em lote

Para usar um filtro de vocabulário personalizado com uma transcrição em lote, veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Trabalhos de transcrição e selecione Criar tarefa (canto superior direito). Isso abre a página Especificar detalhes do trabalho.

Specify job details Info

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type Info

Choose the type of model to use for the transcription job.

General model

To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model

To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language Info

If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification Info

If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

► **Additional settings**

Dê um nome ao seu trabalho e especifique sua mídia de entrada. Opcionalmente, inclua outros campos e escolha Avançar.

- Na página Configurar tarefa, no painel Remoção de conteúdo, ative a Filtragem de vocabulário.

Configure job - optional [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Filter selection
The vocabulary filters shown here are based on your language settings. You can choose up to one vocabulary filter per language. You can also [create a new vocabulary filter](#). [↗](#)

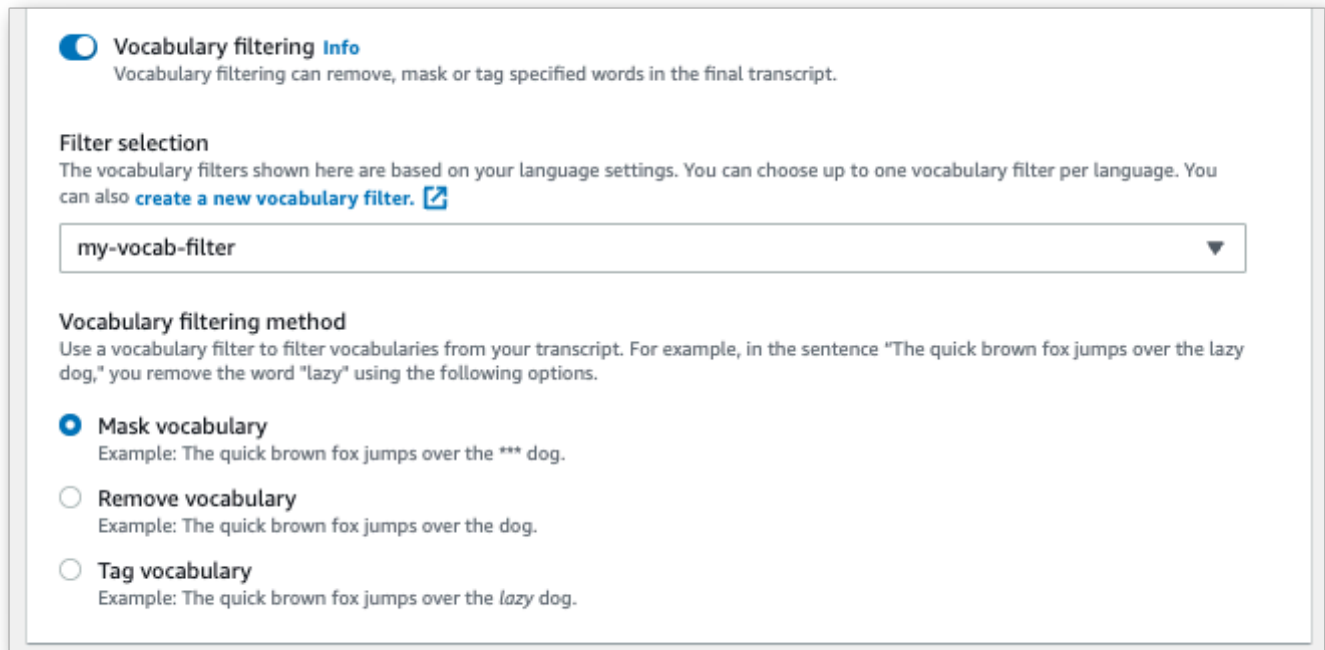
Choose a vocabulary filter ▼

Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

[Cancel](#) [Previous](#) [Create job](#)

4. Selecione seu filtro de vocabulário personalizado no menu suspenso e especifique o método de filtragem.



5. Selecione Criar tarefa para executar sua tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o `Settings` parâmetro com os `VocabularyFilterName` `VocabularyFilterMethod` subparâmetros e. Para obter mais informações, consulte [StartTranscriptionJob](#) e [Settings](#).

```
aws transcribe start-transcription-job \
--region us-west-2 \
--transcription-job-name my-first-transcription-job \
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \
--output-bucket-name DOC-EXAMPLE-BUCKET \
--output-key my-output-files/ \
--language-code en-US \
--settings VocabularyFilterName=my-first-vocabulary-filter,VocabularyFilterMethod=mask
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que inclui seu filtro de vocabulário personalizado com essa tarefa.

```
aws transcribe start-transcription-job \
```

```
--region us-west-2 \  
--cli-input-json file://my-first-vocabulary-filter-job.json
```

O arquivo `my-first-vocabulary-filter-job.json` contém o seguinte corpo da solicitação.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Settings": {  
    "VocabularyFilterName": "my-first-vocabulary-filter",  
    "VocabularyFilterMethod": "mask"  
  }  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para incluir um filtro de vocabulário personalizado usando o `Settings` argumento para o método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#) e [Settings](#).

Para obter exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e serviços cruzados, consulte o [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#) capítulo.

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-transcription-job"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri  
    },  
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',  
    OutputKey = 'my-output-files/',
```

```
LanguageCode = 'en-US',
Settings = {
    'VocabularyFilterName': 'my-first-vocabulary-filter',
    'VocabularyFilterMethod': 'mask'
}
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Usando um filtro de vocabulário personalizado em uma transcrição de streaming


Para usar um filtro de vocabulário personalizado com uma transcrição de streaming, veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Transcrição em tempo real. Role para baixo até Configurações de remoção de conteúdo e expanda esse campo se ele estiver minimizado.

Real-time transcription [Info](#)

See how Amazon Transcribe creates a text copy of speech in real time. Choose **Start streaming** and talk.

Transcription Download full transcript  **Start streaming**

Transcription output Current language: English, US

Choose **Start streaming** to begin a real-time transcription of what you speak into your microphone


00:00 of 15:00 min audio stream

- ▶ **Language settings**
- ▶ **Audio settings**
- ▶ **Content removal settings**
- ▶ **Customizations**

3. Ative a filtragem de vocabulário. Selecione um filtro de vocabulário personalizado no menu suspenso e especifique o método de filtragem.

▼ **Content removal settings**

Vocabulary filtering [Info](#)
Vocabulary filtering removes, masks, or tags words that you specify in your vocabulary filter. Choose a vocabulary filter to see an example.

Filter selection
The vocabulary filters shown here are based on your language settings. You can choose up to one vocabulary filter per language. You can also [create a new vocabulary filter](#).


my-vocab-filter ▼

Vocabulary filtering method [Info](#)
Use a vocabulary filter to filter vocabularies from your transcript. For example, in the sentence "The quick brown fox jumps over the lazy dog," you remove the word "lazy" using the following options.

- Mask vocabulary**
Example: The quick brown fox jumps over the *** dog.
- Remove vocabulary**
Example: The quick brown fox jumps over the dog.
- Tag vocabulary**
Example: The quick brown fox jumps over the *lazy* dog.

Inclua as outras configurações que deseja aplicar à sua transmissão.

4. Agora está tudo pronto para transcrever sua transmissão. Selecione Iniciar transmissão e começar a falar. Para finalizar seu ditado, selecione Interromper streaming.

Fluxo do HTTP/2

Este exemplo cria uma solicitação HTTP/2 que inclui seu filtro de vocabulário personalizado e seu método de filtro. Para obter mais informações sobre como usar o streaming HTTP/2 com Amazon Transcribe, consulte [Configurar um fluxo de HTTP/2](#). Para obter mais detalhes sobre parâmetros e cabeçalhos específicos para Amazon Transcribe, consulte [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-vocabulary-filter-name: my-first-vocabulary-filter
x-amzn-transcribe-vocabulary-filter-method: mask
transfer-encoding: chunked
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

WebSocket riacho

Este exemplo cria um URL predefinido que aplica seu filtro de vocabulário personalizado a um WebSocket stream. Quebras de linha foram adicionadas para uma melhor leitura. Para obter mais informações sobre como usar WebSocket streams com Amazon Transcribe, consulte [Configurando um WebSocket stream](#). Para obter mais detalhes sobre os parâmetros, consulte [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
```

```
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-  
west-2%2Ftranscribe%2Faws4_request  
&X-Amz-Date=20220208T235959Z  
&X-Amz-Expires=300  
&X-Amz-Security-Token=security-token  
&X-Amz-Signature=string  
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date  
&language-code=en-US  
&media-encoding=flac  
&sample-rate=16000  
&vocabulary-filter-name=my-first-vocabulary-filter  
&vocabulary-filter-method=mask
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Detectando fala tóxica

A detecção de fala tóxica foi projetada para ajudar a moderar plataformas de mídia social que envolvem peer-to-peer diálogo, como jogos online e plataformas de bate-papo social. O uso de fala tóxica pode ser profundamente prejudicial para indivíduos, grupos de colegas e comunidades. Sinalizar linguagem prejudicial ajuda as organizações a manter as conversas civilizadas e a manter um ambiente on-line seguro e inclusivo para que os usuários criem, compartilhem e participem livremente.

Amazon Transcribe A detecção de toxicidade utiliza sinais baseados em áudio e texto para identificar e classificar conteúdo tóxico baseado em voz em sete categorias, incluindo assédio sexual, discurso de ódio, ameaça, abuso, palavrões, insultos e gráficos. Além do texto, Amazon Transcribe A detecção de toxicidade usa sinais de fala, como tons e tons, para aprimorar a intenção tóxica da fala. Essa é uma melhoria em relação aos sistemas padrão de moderação de conteúdo, projetados para se concentrar apenas em termos específicos, sem levar em conta a intenção.

Amazon Transcribe sinaliza e categoriza a fala tóxica, o que minimiza o volume de dados que devem ser processados manualmente. Isso permite que os moderadores de conteúdo gerenciem com rapidez e eficiência o discurso em suas plataformas.

As categorias de fala tóxica incluem:

- **Profanidade:** Discurso que contém palavras, frases ou acrônimos indelicados, vulgares ou ofensivos.
- **Discurso de ódio:** Discurso que critica, insulta, denuncia ou desumaniza uma pessoa ou grupo com base em uma identidade (como raça, etnia, gênero, religião, orientação sexual, habilidade e origem nacional).
- **Sexual:** Discurso que indica interesse sexual, atividade ou excitação usando referências diretas ou indiretas a partes do corpo, características físicas ou sexo.
- **Insultos:** Discurso que inclui linguagem humilhante, humilhante, zombeteira, insultante ou depreciativa. Esse tipo de linguagem também é rotulado como bullying.
- **Violência ou ameaça:** Discurso que inclui ameaças que buscam infligir dor, ferimento ou hostilidade a uma pessoa ou grupo.
- **Gráfico:** Fala que usa imagens visualmente descritivas e desagradavelmente vívidas. Esse tipo de linguagem geralmente é intencionalmente prolixo para ampliar o desconforto do destinatário.

- **Assédio ou abuso:** Discurso destinado a afetar o bem-estar psicológico do destinatário, incluindo termos humilhantes e objetivadores. Esse tipo de linguagem também é rotulado como assédio.

A detecção de toxicidade analisa os segmentos da fala (a fala entre pausas naturais) e atribui pontuações de confiança a esses segmentos. Os escores de confiança são valores entre 0 e 1. Uma pontuação de confiança maior indica uma maior probabilidade de que o conteúdo seja uma fala tóxica na categoria associada. Você pode usar essas pontuações de confiança para definir o limite apropriado de detecção de toxicidade para seu caso de uso.

Note

A detecção de toxicidade só está disponível para transcrições em lote em inglês dos EUA(en-US).

Visualizar [exemplo de saída](#) no formato JSON.

Usando a detecção de fala tóxica

Usando detecção de fala tóxica em uma transcrição em lote

Para usar a detecção de fala tóxica com uma transcrição em lote, veja exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha **Trabalhos de transcrição** e, em seguida, selecione **Criar emprego** (canto superior direito). Isso abre o **Especifique os detalhes do trabalho** página.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)

Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings

You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language

Choose the language of the input audio.

► **Additional settings**

3. Sobre oEspecifique os detalhes do trabalhopágina, você também pode ativar a redação de PII, se quiser. Observe que as outras opções listadas não são compatíveis com a detecção de toxicidade. Selecione Próximo. Isso leva você para oConfigurar tarefa - opcionalpágina. NoConfigurações de áudiopainel, selecioneDetecção de toxicidade.

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or partition speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Toxicity detection [Info](#)
Flag toxic speech in your transcription output

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)
Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)
A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

[Cancel](#) [Previous](#) [Create job](#)

4. Selecionar Criar emprego para executar seu trabalho de transcrição.
5. Quando seu trabalho de transcrição estiver concluído, você poderá fazer o download da transcrição no menu suspenso na página de detalhes do trabalho de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e `ToxicityDetection` parâmetro. Para ter mais informações, consulte [StartTranscriptionJob](#) e [ToxicityDetection](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--toxicity-detection ToxicityCategories=ALL
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que inclui detecção de toxicidade.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-first-toxicity-job.json
```

O arquivo `my-first-toxicity-job.json` contém o seguinte corpo da solicitação.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "ToxicityDetection": [  
    {  
      "ToxicityCategories": [ "ALL" ]  
    }  
  ]  
}
```


AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para habilitar `ToxicityDetection` para o `start_transcription_job` método. Para ter mais informações, consulte [StartTranscriptionJob](#) e [ToxicityDetection](#).

Para obter exemplos adicionais usando os SDKs AWS, incluindo exemplos específicos de recursos, cenários e serviços cruzados, referem-se ao [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#) capítulo.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    ToxicityDetection = [
        {
            'ToxicityCategories': ['ALL']
        }
    ]
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Exemplo de saída

A fala tóxica é marcada e categorizada em sua saída de transcrição. Cada instância de fala tóxica é categorizada e atribuída a uma pontuação de confiança (um valor entre 0 e 1). Um valor de confiança maior indica uma maior probabilidade de que o conteúdo seja uma fala tóxica dentro da categoria especificada.

Exemplo de saída (JSON)

A seguir está um exemplo de saída no formato JSON que mostra fala tóxica categorizada com pontuações de confiança associadas.

```
{
  "jobName": "my-toxicity-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [...],
    "items": [...],
    "toxicity_detection": [
      {
        "text": "What the * are you doing man? That's why I didn't want to play
with your * . man it was a no, no I'm not calming down * man. I well I spent I spent
too much * money on this game.",
        "toxicity": 0.7638,
        "categories": {
          "profanity": 0.9913,
          "hate_speech": 0.0382,
          "sexual": 0.0016,
          "insult": 0.6572,
          "violence_or_threat": 0.0024,
          "graphic": 0.0013,
          "harassment_or_abuse": 0.0249
        },
        "start_time": 8.92,
        "end_time": 21.45
      },
      Items removed for brevity
    ],
    {
      "text": "What? Who? What the * did you just say to me? What's your
address? What is your * address? I will pull up right now on your * * man. Take your *
back to , tired of this **.",
      "toxicity": 0.9816,
```

```
        "categories": {
            "profanity": 0.9865,
            "hate_speech": 0.9123,
            "sexual": 0.0037,
            "insult": 0.5447,
            "violence_or_threat": 0.5078,
            "graphic": 0.0037,
            "harassment_or_abuse": 0.0613
        },
        "start_time": 43.459,
        "end_time": 54.639
    },
]
},
...
"status": "COMPLETED"
}
```


Editar PII no trabalho em lote

Ao redigir informações de identificação pessoal (PII) de uma transcrição durante um trabalho de transcrição em lote, Amazon Transcribe substitui cada instância identificada de PII pelo corpo do texto principal da sua transcrição. [PII] Você também pode ver o tipo de PII que está redigido na word-for-word parte da saída da transcrição. Para obter um exemplo de saída, consulte [Exemplo de saída editada \(lote\)](#).

A redação com transcrições em lote está disponível em inglês dos EUA (en-US) e espanhol dos EUA (es-US). A edição não é compatível com a [identificação de idioma](#).

Tanto as transcrições editadas quanto as não editadas são armazenadas no mesmo bucket de saída. Amazon S3 Amazon Transcribe os armazena em um bucket especificado por você ou no Amazon S3 bucket padrão gerenciado pelo serviço.

Tipos de PII que Amazon Transcribe podem ser reconhecidos para transcrições em lote

Tipo de PII	Descrição
ADDRESS	Um endereço físico, como 100 Main Street, Anytown, USA ou Suite #12, Building 123. Um endereço pode incluir rua, prédio, local, cidade, estado, país, condado, CEP, jurisdição, bairro e muito mais.
ALL	Reduza ou identifique todos os tipos de PII listados nesta tabela.
BANK_ACCOUNT_NUMBER	Um número de conta bancária dos EUA. Normalmente, eles têm entre 10 e 12 dígitos, mas o Amazon Transcribe também reconhece números de contas bancárias quando somente os últimos quatro dígitos estão presentes.
BANK_ROUTING	Um número de roteamento de conta bancária dos EUA. Normalmente, eles têm nove dígitos, mas o Amazon Transcribe também reconhece números de roteamento quando apenas os últimos quatro dígitos estão presentes.

Tipo de PII	Descrição
CREDIT_DEBIT_CVV	Um código de verificação de cartão (CVV) de 3 dígitos que está presente nos cartões de crédito e débito VISA e Discover. MasterCard Nos cartões de crédito ou débito da American Express, é um código numérico de quatro dígitos.
CREDIT_DEBIT_EXPIRY	Data de validade do cartão de crédito ou débito. Esse número geralmente tem quatro dígitos e é formatado como mês/ano ou MM/AA. Por exemplo, Amazon Transcribe pode reconhecer datas de vencimento como 21/01, 01/2021 e janeiro de 2021.
CREDIT_DEBIT_NUMBER	O número de um cartão de crédito ou débito. Esses números podem variar de 13 a 16 dígitos, mas Amazon Transcribe também reconhece números de cartão de crédito ou débito quando somente os últimos 4 dígitos estão presentes.
EMAIL	Um endereço de e-mail, como efua.owusu@email.com.
NAME	O nome de uma pessoa. Esse tipo de entidade não inclui títulos, como Sr., Sra., Senhorita ou Dr.. Não Amazon Transcribe aplica esse tipo de entidade a nomes que fazem parte de organizações ou endereços. Por exemplo, Amazon Transcribe reconhece a John Doe Organization como uma organização e a Jane Doe Street como um endereço.
PHONE	Um número de telefone. Esse tipo de entidade também inclui números de fax e pager.

Tipo de PII	Descrição
PIN	Um número de identificação pessoal (PIN) de quatro dígitos que permite que alguém acesse as informações da conta bancária.
SSN	Um Número de Seguro Social (SSN) é um número de 9 dígitos emitido para cidadãos dos EUA, residentes permanentes e residentes temporários que trabalham. Amazon Transcribe também reconhece números de previdência social quando apenas os últimos 4 dígitos estão presentes.

Você pode iniciar um trabalho de transcrição em lote usando o AWS Management Console AWS CLI, ou AWS SDK.

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Tarefas de transcrição e selecione Criar tarefa (no canto superior direito). Isso abrirá a página Especificar os detalhes da tarefa.
3. Depois de preencher os campos desejados na página Especificar os detalhes da tarefa, selecione Próximo para ir até a página Configurar tarefa - opcional. Ali você encontrará o painel de Remoção de conteúdo com o botão Edição de PII.

Configure job - *optional* [Info](#)

Audio settings

Audio identification [Info](#)
Choose to split multi-channel audio into separate channels for transcription, or identify speakers in the input audio.

Alternative results [Info](#)
Enable to view more transcription results

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

4. Depois de selecionar Edição de PII, você tem a opção de escolher todos os tipos de PII que deseja editar. Você também pode optar por ter uma transcrição não editada se selecionar a caixa Incluir transcrição sem ocultação na saída da tarefa.

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)
 Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Include unredacted transcript in job output
 Returns unredacted version of the transcript in addition to the redacted version.

Select PII entity types (11 of 11 selected)

Select All

Financial (6 of 6 selected)

<input checked="" type="checkbox"/> BANK_ACCOUNT_NUMBER	<input checked="" type="checkbox"/> BANK_ROUTING	<input checked="" type="checkbox"/> CREDIT_DEBIT_NUMBER
<input checked="" type="checkbox"/> CREDIT_DEBIT_CVV	<input checked="" type="checkbox"/> CREDIT_DEBIT_EXPIRY	<input checked="" type="checkbox"/> PIN

Personal (5 of 5 selected)

<input checked="" type="checkbox"/> NAME	<input checked="" type="checkbox"/> ADDRESS	<input checked="" type="checkbox"/> PHONE
<input checked="" type="checkbox"/> EMAIL	<input checked="" type="checkbox"/> SSN	

Vocabulary filtering [Info](#)
 Vocabulary filtering can remove, mask or tag specified words in the final transcript.

5. Selecione Criar tarefa para executar a tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o `content-redaction` parâmetro. Para obter mais informações, consulte [StartTranscriptionJob](#) e [ContentRedaction](#).

```
aws transcribe start-transcription-job \
--region us-west-2 \
--transcription-job-name my-first-transcription-job \
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \
--output-bucket-name DOC-EXAMPLE-BUCKET \
--output-key my-output-files/ \
--language-code en-US \
```

```
--content-redaction
RedactionType=PII,RedactionOutput=redacted,PiiEntityTypes=NAME,ADDRESS,BANK_ACCOUNT_NUMBER
```

Aqui está outro exemplo usando o [start-transcription-job](#) método, e o corpo da solicitação redige as PII desse trabalho.

```
aws transcribe start-transcription-job \
--region us-west-2 \
--cli-input-json file://filepath/my-first-redaction-job.json
```

O arquivo `my-first-redaction-job.json` contém o seguinte corpo da solicitação.

```
{
  "TranscriptionJobName": "my-first-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "LanguageCode": "en-US",
  "ContentRedaction": {
    "RedactionOutput": "redacted",
    "RedactionType": "PII",
    "PiiEntityTypes": [
      "NAME",
      "ADDRESS",
      "BANK_ACCOUNT_NUMBER"
    ]
  }
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para redigir conteúdo usando o `ContentRedaction` argumento do método [start_transcription_job](#). Para obter mais informações, consulte [StartTranscriptionJob](#) e [ContentRedaction](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_transcription_job(
    TranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    ContentRedaction = {
        'RedactionOutput': 'redacted',
        'RedactionType': 'PII',
        'PiiEntityTypes': [
            'NAME', 'ADDRESS', 'BANK_ACCOUNT_NUMBER'
        ]
    }
)

while True:
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Note

A redação de PII para trabalhos em lote só é suportada nestas áreas Regiões da AWS: Ásia-Pacífico (Hong Kong), Ásia-Pacífico (Mumbai), Ásia-Pacífico (Seul), Ásia-Pacífico (Cingapura), Ásia-Pacífico (Sydney), Ásia-Pacífico (Tóquio), GovCloud (Oeste dos EUA), Canadá (Central), UE (Frankfurt), UE (Irlanda), UE (Londres), UE (Paris), Oriente Médio (Bahrein), América do Sul (São Paulo), Leste dos EUA (Norte da Virgínia), Leste dos EUA (Ohio), Oeste dos EUA (Oregon) e Oeste dos EUA (Norte da Califórnia).

Editar ou identificar PII em um fluxo em tempo real

Ao editar informações de identificação pessoal (PII) em uma transcrição de streaming, o Amazon Transcribe substitui cada caso de PII identificado por [PII] na transcrição.

Uma opção adicional disponível para transcrições de streaming é a identificação de PII. Quando você ativa a identificação de PII, Amazon Transcribe rotula as PII nos resultados da transcrição sob um objeto. `Entities` Para obter um exemplo de saída, consulte [Exemplo de saída de streaming editada](#) e [Exemplo de saída de identificação PII](#).

A redação e identificação de PII com transcrições de streaming estão disponíveis nos seguintes dialetos em inglês: australiano (en-AU), britânico (), EUA (en-GB) e espanhol (dialeto americanoen-US). es-US

A identificação e a edição de PII para trabalhos de streaming são realizadas somente após a transcrição completa dos segmentos de áudio.

Tipos de PII que Amazon Transcribe podem ser reconhecidos para transcrições de streaming

Tipo de PII	Descrição
ADDRESS	Um endereço físico, como 100 Main Street, Anytown, USA ou Suite #12, Building 123. Um endereço pode incluir rua, prédio, local, cidade, estado, país, condado, CEP, jurisdição, bairro e muito mais.
ALL	Reduza ou identifique todos os tipos de PII listados nesta tabela.
BANK_ACCOUNT_NUMBER	Um número de conta bancária dos EUA. Normalmente, eles têm entre 10 e 12 dígitos, mas o Amazon Transcribe também reconhece números de contas bancárias quando somente os últimos quatro dígitos estão presentes.
BANK_ROUTING	Um número de roteamento de conta bancária dos EUA. Normalmente, eles têm nove dígitos, mas o Amazon Transcribe também reconhece

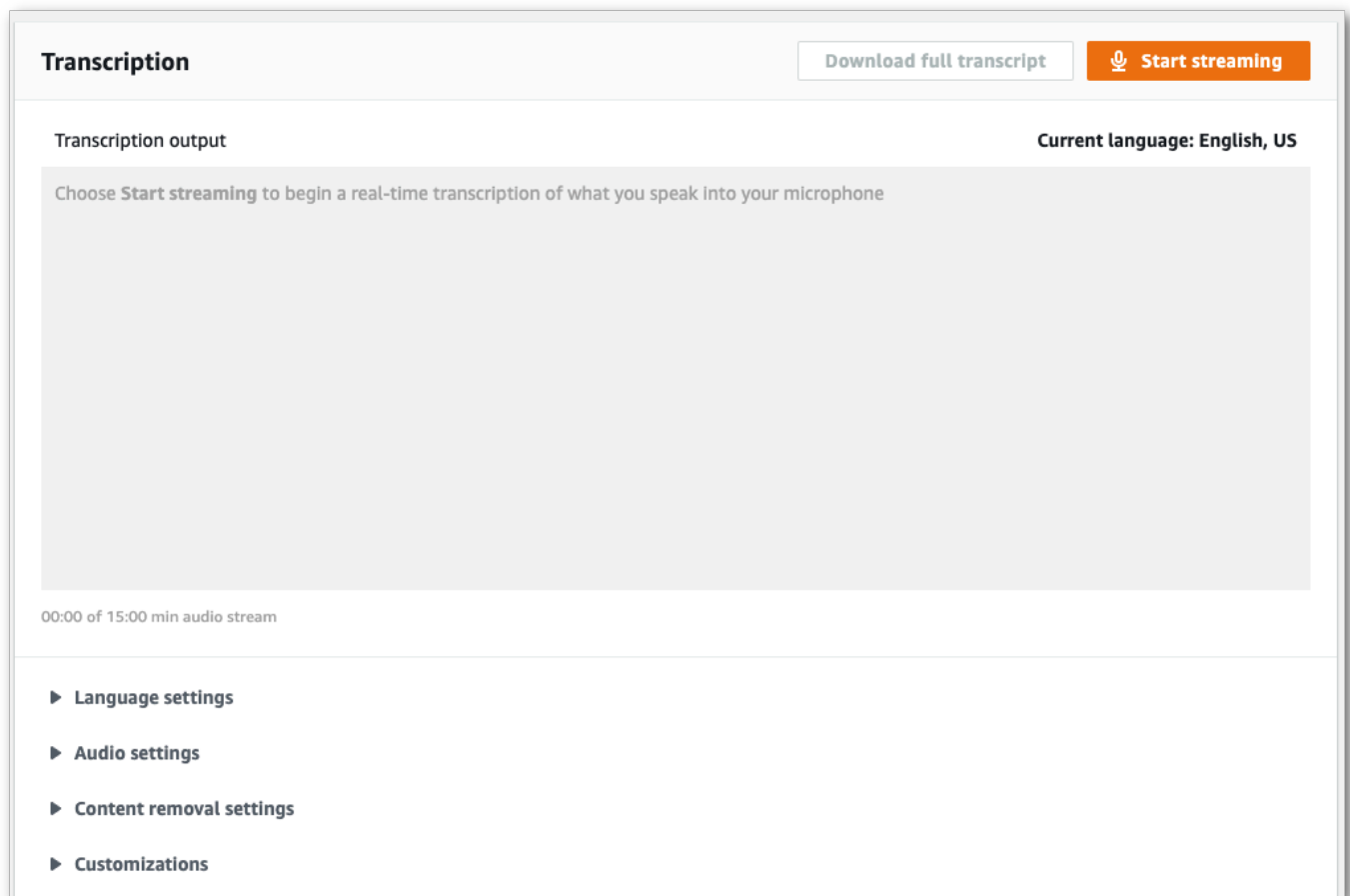
Tipo de PII	Descrição
	números de roteamento quando apenas os últimos quatro dígitos estão presentes.
CREDIT_DEBIT_CVV	Um código de verificação de cartão (CVV) de 3 dígitos que está presente nos cartões de crédito e débito VISA e Discover. MasterCard Nos cartões de crédito ou débito da American Express, é um código numérico de quatro dígitos.
CREDIT_DEBIT_EXPIRY	Data de validade do cartão de crédito ou débito. Esse número geralmente tem quatro dígitos e é formatado como mês/ano ou MM/AA. Por exemplo, Amazon Transcribe pode reconhecer datas de vencimento como 21/01, 01/2021 e janeiro de 2021.
CREDIT_DEBIT_NUMBER	O número de um cartão de crédito ou débito. Esses números podem variar de 13 a 16 dígitos, mas Amazon Transcribe também reconhece números de cartão de crédito ou débito quando somente os últimos 4 dígitos estão presentes.
EMAIL	Um endereço de e-mail, como efua.owusu@email.com.
NAME	O nome de uma pessoa. Esse tipo de entidade não inclui títulos, como Sr., Sra., Senhorita ou Dr.. Não Amazon Transcribe aplica esse tipo de entidade a nomes que fazem parte de organizações ou endereços. Por exemplo, Amazon Transcribe reconhece a John Doe Organization como uma organização e a Jane Doe Street como um endereço.

Tipo de PII	Descrição
PHONE	Um número de telefone. Esse tipo de entidade também inclui números de fax e pager.
PIN	Um número de identificação pessoal (PIN) de quatro dígitos que permite que alguém acesse as informações da conta bancária.
SSN	Um Número de Seguro Social (SSN) é um número de 9 dígitos emitido para cidadãos dos EUA, residentes permanentes e residentes temporários que trabalham. Amazon Transcribe também reconhece números de previdência social quando apenas os últimos 4 dígitos estão presentes.

Você pode iniciar uma transcrição de streaming usando o AWS Management Console WebSocket, ou HTTP/2.

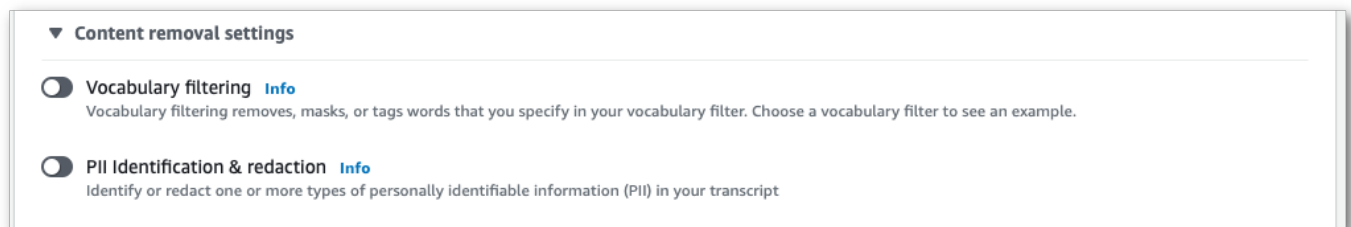
AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Transcrição em tempo real. Role para baixo até Configurações de remoção de conteúdo e expanda esse campo se ele estiver minimizado.



The screenshot shows the Amazon Transcribe transcription interface. At the top left, the word "Transcription" is displayed. To its right are two buttons: "Download full transcript" and "Start streaming" (which is highlighted in orange). Below this is the "Transcription output" section, which includes the text "Choose Start streaming to begin a real-time transcription of what you speak into your microphone" and a large grey rectangular area for the transcription. To the right of this area, it says "Current language: English, US". Below the transcription area, a progress indicator shows "00:00 of 15:00 min audio stream". At the bottom, there is a list of settings with expandable arrows: "Language settings", "Audio settings", "Content removal settings", and "Customizations".

3. Ative a Identificação e ocultação de PII.



The screenshot shows the "Content removal settings" section in the Amazon Transcribe interface. It is titled "Content removal settings" with a downward arrow. There are two radio button options, both of which are currently unselected:

- Vocabulary filtering** [Info](#)
Vocabulary filtering removes, masks, or tags words that you specify in your vocabulary filter. Choose a vocabulary filter to see an example.
- PII Identification & redaction** [Info](#)
Identify or redact one or more types of personally identifiable information (PII) in your transcript

4. Selecione Somente identificação ou Identificação e ocultação e escolha os tipos de entidade de PII que você deseja identificar ou editar na transcrição.

Content removal settings

Vocabulary filtering [Info](#)
Vocabulary filtering removes, masks, or tags words that you specify in your vocabulary filter. Choose a vocabulary filter to see an example.

PII Identification & redaction [Info](#)
Identify or redact one or more types of personally identifiable information (PII) in your transcript

Select PII detection type

Identification only
Label the type of PII identified but not redact it in the transcription output

Identification & redaction
Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123)456-7890 will be masked as [PHONE]

Select PII entity types (22 of 22 selected)

Select All

Financial (6 of 6 selected)

<input checked="" type="checkbox"/> BANK_ACCOUNT_NUMBER	<input checked="" type="checkbox"/> BANK_ROUTING	<input checked="" type="checkbox"/> CREDIT_DEBIT_NUMBER
<input checked="" type="checkbox"/> CREDIT_DEBIT_CVV	<input checked="" type="checkbox"/> CREDIT_DEBIT_EXPIRY	<input checked="" type="checkbox"/> PIN

Personal (8 of 8 selected)

<input checked="" type="checkbox"/> NAME	<input checked="" type="checkbox"/> ADDRESS	<input checked="" type="checkbox"/> PHONE
<input checked="" type="checkbox"/> EMAIL	<input checked="" type="checkbox"/> SSN	<input checked="" type="checkbox"/> PASSPORT_NUMBER
<input checked="" type="checkbox"/> DRIVER_ID	<input checked="" type="checkbox"/> AGE	

Digital footprint (7 of 7 selected)

<input checked="" type="checkbox"/> URL	<input checked="" type="checkbox"/> USERNAME	<input checked="" type="checkbox"/> PASSWORD
<input checked="" type="checkbox"/> AWS_ACCESS_KEY	<input checked="" type="checkbox"/> AWS_SECRET_KEY	<input checked="" type="checkbox"/> IP_ADDRESS
<input checked="" type="checkbox"/> MAC_ADDRESS		

Other (1 of 1 selected)

DATE_TIME

► Customizations

5. Agora está tudo pronto para transcrever seu fluxo. Selecione Iniciar transmissão e comece a falar. Para finalizar o ditado, selecione Interromper transmissão.

WebSocket stream

Este exemplo cria uma URL pré-assinada que usa a redação de PII (ou identificação de PII) em um stream. WebSocket Quebras de linha foram adicionadas para uma melhor leitura. Para obter mais informações sobre o uso de WebSocket fluxos com Amazon Transcribe, consulte [Configurando um WebSocket stream](#). Para obter mais detalhes sobre os parâmetros, consulte [StartStreamTranscription](#).

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/stream-transcription-
websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
```



```
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&language-code=en-US
&media-encoding=flac
&sample-rate=16000
&pii-entity-types=NAME,ADDRESS
&content-redaction-type=PII (or &content-identification-type=PII)
```

Não é possível usar `content-identification-type` e `content-redaction-type` na mesma solicitação.

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Fluxo HTTP/2

Este exemplo cria uma solicitação HTTP/2 com a identificação de PII ou a edição de PII habilitada. Para obter mais informações sobre como usar o streaming HTTP/2 com Amazon Transcribe, consulte [Configurar um fluxo de HTTP/2](#). Para obter mais detalhes sobre parâmetros e cabeçalhos específicos de Amazon Transcribe, consulte [StartStreamTranscription](#).

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-content-identification-type: PII (or x-amzn-transcribe-content-
redaction-type: PII)
x-amzn-transcribe-pii-entity-types: NAME,ADDRESS
transfer-encoding: chunked
```

Não é possível usar `content-identification-type` e `content-redaction-type` na mesma solicitação.

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Note

A redação de PII para streaming só é suportada nestas áreas Regiões da AWS: Ásia-Pacífico (Seul), Ásia-Pacífico (Sydney), Ásia-Pacífico (Tóquio), Canadá (Central), UE (Frankfurt), UE (Irlanda), UE (Londres), Leste dos EUA (Norte da Virgínia), Leste dos EUA (Ohio) e Oeste dos EUA (Oregon).

Exemplo de saída de redação e identificação de PII

Os exemplos a seguir mostram a saída editada de trabalhos em lote e de streaming e a identificação de PII de uma tarefa de streaming.

Os trabalhos de transcrição usando redação de conteúdo geram dois tipos de confiança valores. A confiança do reconhecimento automático de fala (ASR) indica os itens que têm o `type` de `pronunciation` ou `punctuation` como uma expressão específica. Na saída de transcrição a seguir, a palavra `Good` tem um `confidence` de `1.0`. Esse valor de confiança indica que Amazon Transcribe está 100% confiante de que a palavra proferida nesta transcrição é “Boa”. O valor de `confidence` para uma tag `[PII]` é a confiança de que o discurso sinalizado para a redação realmente é PII. Na saída da transcrição a seguir, o `confidence` de `0.9999` indica que Amazon Transcribe está 99,99 por cento confiante de que a entidade redigida na transcrição é PII.

Exemplo de saída editada (lote)

```
{
  "jobName": "my-first-transcription-job",
  "accountId": "111122223333",
  "isRedacted": true,
  "results": {
    "transcripts": [
      {
        "transcript": "Good morning, everybody. My name is [PII], and today I
feel like
```

```

    sharing a whole lot of personal information with you. Let's start with
my Social
    Security number [PII]. My credit card number is [PII] and my C V V code
is [PII].
    I hope that Amazon Transcribe is doing a good job at redacting that
personal
    information away. Let's check."
    }
  ],
  "items": [
    {
      "start_time": "2.86",
      "end_time": "3.35",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "Good"
        }
      ],
      "type": "pronunciation"
    },
    Items removed for brevity
    {
      "start_time": "5.56",
      "end_time": "6.25",
      "alternatives": [
        {
          "content": "[PII]",
          "redactions": [
            {
              "confidence": "0.9999",
              "type": "NAME",
              "category": "PII"
            }
          ]
        }
      ],
      "type": "pronunciation"
    },
    Items removed for brevity
  ],
  "status": "COMPLETED"

```

}

Aqui está a transcrição não editada para comparação:

```
{
  "jobName": "job id",
  "accountId": "111122223333",
  "isRedacted": false,
  "results": {
    "transcripts": [
      {
        "transcript": "Good morning, everybody. My name is Mike, and today I
feel like
my Social
job
at redacting that personal information away. Let's check."
      }
    ],
    "items": [
      {
        "start_time": "2.86",
        "end_time": "3.35",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "Good"
          }
        ],
        "type": "pronunciation"
      },
      Items removed for brevity
      {
        "start_time": "5.56",
        "end_time": "6.25",
        "alternatives": [
          {
            "confidence": "0.9999",
            "content": "Mike",
          }
        ],
      }
    ]
  }
}
```

```

        "type": "pronunciation"
      },
      Items removed for brevity
    ],
  },
  "status": "COMPLETED"
}

```

Exemplo de saída de streaming editada

```

{
  "TranscriptResultStream": {
    "TranscriptEvent": {
      "Transcript": {
        "Results": [
          {
            "Alternatives": [
              {
                "Transcript": "my name is [NAME]",
                "Items": [
                  {
                    "Content": "my",
                    "EndTime": 0.3799375,
                    "StartTime": 0.0299375,
                    "Type": "pronunciation"
                  },
                  {
                    "Content": "name",
                    "EndTime": 0.5899375,
                    "StartTime": 0.3899375,
                    "Type": "pronunciation"
                  },
                  {
                    "Content": "is",
                    "EndTime": 0.7899375,
                    "StartTime": 0.5999375,
                    "Type": "pronunciation"
                  },
                  {
                    "Content": "[NAME]",
                    "EndTime": 1.0199375,
                    "StartTime": 0.7999375,
                    "Type": "pronunciation"
                  }
                ]
              }
            ]
          }
        ]
      }
    }
  }
}

```



```
}
```


Criar legendas de vídeo

Amazon Transcribe suporta saídas WebVTT (*.vtt) e SubRip (*.srt) para uso como legendas de vídeo. Você pode selecionar um ou ambos os tipos de arquivo ao configurar seu trabalho de transcrição de vídeo em lote. Ao usar o recurso de legenda, os arquivos de legenda selecionados e um arquivo de transcrição normal (contendo informações adicionais) são produzidos. Os arquivos de legenda e transcrição são enviados para o mesmo destino.

As legendas são exibidas ao mesmo tempo em que o texto é falado e permanecem visíveis até que haja uma pausa natural ou o alto-falante termine de falar. Observe que se você habilitar as legendas em sua solicitação de transcrição e seu áudio não contiver voz, um arquivo de legenda não será criado.

Important

Amazon Transcribe usa um índice inicial padrão de 0 para saída de legendas, que difere do valor mais amplamente usado de 1. Se você precisar de um índice inicial de 1, você pode especificar isso na AWS Management Console ou na sua solicitação de API usando o [OutputStartIndex](#) parâmetro.

Usar o índice inicial incorreto pode resultar em erros de compatibilidade com outros serviços, portanto, verifique qual índice inicial você precisa antes de criar suas legendas. Se você não tiver certeza de qual valor usar, recomendamos escolher 1. Consulte [Subtitles](#) para obter mais informações.

Recursos compatíveis com legendas:

- Redação de conteúdo — Qualquer conteúdo editado é refletido como 'PII' nos arquivos de saída da legenda e da transcrição normal. O áudio não foi alterado.
- Filtros de vocabulário — Os arquivos de legenda são gerados a partir do arquivo de transcrição, portanto, todas as palavras que você filtrar na saída padrão da transcrição também são filtradas nas legendas. O conteúdo filtrado é exibido como espaço em branco ou *** em seus arquivos de transcrição e legenda. O áudio não foi alterado.
- Diarização do alto-falante — Se houver vários alto-falantes em um determinado segmento de legenda, traços serão usados para distinguir cada alto-falante. Isso se aplica tanto ao WebVTT quanto aos SubRip formatos; por exemplo:

- -- Texto falado pela Pessoa 1
- -- Texto falado pela Pessoa 2

Os arquivos de legenda são armazenados no mesmo Amazon S3 local da saída da transcrição.

Consulte [Amazon Transcribe Video Snacks: Criando legendas de vídeo sem escrever nenhum código](#) para ver um vídeo passo a passo sobre a criação de legendas.

Geração de arquivos de legendas

Você pode criar arquivos de legendas usando o AWS Management Console, AWS CLI, ou AWSSDKs; veja os exemplos a seguir:

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Trabalhos de transcrição e selecione Criar trabalho (canto superior direito). Isso abre a página Especificar detalhes do trabalho. As opções de legenda estão localizadas no painel Dados de saída.
3. Selecione os formatos que você deseja para seus arquivos de legendas e, em seguida, escolha um valor para seu índice inicial. Observe que o Amazon Transcribe padrão é 0, mas 1 é mais amplamente usado. Se você não tiver certeza de qual valor usar, recomendamos escolher 1, pois isso pode melhorar a compatibilidade com outros serviços.

Output data

Output data location type info [Info](#)

Service-managed S3 bucket
The output will be removed after 90 days when the job expires.

Customer specified S3 bucket
The output will not be removed from bucket even after the job expires.

Subtitle file format [Info](#)

SRT (SubRip)

VTT (WebVTT)

Specify the start index

0 ▼

4. Preencha todos os outros campos que você deseja incluir na página Especificar detalhes do trabalho e selecione Avançar. Isso leva você à página Configurar tarefa - opcional.
5. Selecione Criar tarefa para executar sua tarefa de transcrição.

AWS CLI

Este exemplo usa o [start-transcription-job](#) comando e o Subtitles parâmetro. Para ter mais informações, consulte [StartTranscriptionJob](#) e [Subtitles](#).

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--transcription-job-name my-first-transcription-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--output-key my-output-files/ \  
--language-code en-US \  
--subtitles Formats=vtt,srt,OutputStartIndex=1
```

Aqui está outro exemplo usando o [start-transcription-job](#) comando e um corpo de solicitação que adiciona legendas a esse trabalho.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://my-first-subtitle-job.json
```

O arquivo *my-first-subtitle-job.json* contém o corpo da solicitação a seguir.

```
{  
  "TranscriptionJobName": "my-first-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Subtitles": {  
    "Formats": [  
      "vtt", "srt"  
    ],  
    "OutputStartIndex": 1  
  }  
}
```

```
}  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para adicionar legendas usando o `Subtitles` argumento do método [start_transcription_job](#). Para ter mais informações, consulte [StartTranscriptionJob](#) e [Subtitles](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

```
from __future__ import print_function  
import time  
import boto3  
transcribe = boto3.client('transcribe', 'us-west-2')  
job_name = "my-first-transcription-job"  
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
transcribe.start_transcription_job(  
    TranscriptionJobName = job_name,  
    Media = {  
        'MediaFileUri': job_uri  
    },  
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',  
    OutputKey = 'my-output-files/',  
    LanguageCode = 'en-US',  
    Subtitles = {  
        'Formats': [  
            'vtt', 'srt'  
        ],  
        'OutputStartIndex': 1  
    }  
)  
  
while True:  
    status = transcribe.get_transcription_job(TranscriptionJobName = job_name)  
    if status['TranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED', 'FAILED']:  
        break  
    print("Not ready yet...")  
    time.sleep(5)  
print(status)
```

Analisar o áudio da central de atendimento com o Call Analytics

Use o Amazon Transcribe Call Analytics para obter informações sobre as interações entre clientes e agentes. O Call Analytics foi projetado especificamente para o áudio da central de atendimento e fornece automaticamente dados valiosos relacionados a cada chamada e a cada participante. Também é possível aprimorar os dados em pontos específicos durante a chamada. Por exemplo, é possível comparar o sentimento do cliente nos primeiros segundos de uma chamada com o último quarto da chamada para ver se o atendente proporcionou uma experiência positiva. Outros exemplos de casos de uso estão listados na [seção a seguir](#).

O Call Analytics está disponível para transcrições pós-chamada e em tempo real. Se você estiver transcrevendo um arquivo localizado em um Amazon S3 bucket, você está realizando uma transcrição pós-chamada. Se estiver transcrevendo um streaming de áudio, isso significa que está realizando uma transcrição em tempo real. Esses dois métodos de transcrição oferecem diferentes informações e recursos do Call Analytics. Para obter mais detalhes sobre cada método, consulte [Análise pós-chamada](#) e [Análise de chamadas em tempo real](#).

Com as transcrições do Call Analytics em tempo real, você também pode incluir [análises pós-chamada](#) na solicitação. Sua transcrição de análise pós-chamada é armazenada no Amazon S3 bucket que você especifica em sua solicitação. Consulte [Análise pós-chamada com transcrições em tempo real](#) para obter mais informações.

Operações de API específicas do Call Analytics

Pós-chamada: [CreateCallAnalyticsCategory](#), [DeleteCallAnalyticsCategory](#), [DeleteCallAnalyticsJob](#), [GetCallAnalyticsCategory](#), [GetCallAnalyticsJob](#), [ListCallAnalyticsCategories](#), [ListCallAnalyticsJobs](#), [StartCallAnalyticsJob](#), [UpdateCallAnalyticsCategory](#)
Em tempo real: [StartCallAnalyticsStreamTranscription](#), [StartCallAnalyticsStreamTranscriptionWebSocket](#)

Casos de uso comuns

Transcrições pós-chamada:

- Monitorar a frequência dos problemas ao longo do tempo: use a [categorização de chamadas](#) para identificar palavras-chave recorrentes nas transcrições.
- Obter informações sobre sua experiência de atendimento ao cliente: use as [características da chamada](#) (tempo sem conversa, tempo de conversa, interrupções, volume da voz, velocidade da conversa) e análise de sentimentos para determinar se os problemas do cliente estão sendo resolvidos adequadamente durante a chamada.
- Garantir a conformidade regulatória ou a adesão à política da empresa: defina [palavras-chave e frases](#) para saudações ou avisos de isenção de responsabilidade específicos da empresa para verificar se os atendentes estão cumprindo os requisitos regulatórios.
- Melhorar o processamento dos dados pessoais dos clientes: use a [edição de PII](#) na saída de transcrição ou no arquivo de áudio para ajudar a proteger a privacidade do cliente.
- Melhorar o treinamento da equipe: use critérios (sentimento, tempo sem conversa, interrupções, velocidade da conversa) para sinalizar transcrições que possam ser usadas como exemplos de interações positivas ou negativas com o cliente.
- Avaliar a eficácia da equipe na criação de uma experiência positiva para o cliente: use a [análise de sentimentos](#) para avaliar se os atendentes conseguem transformar um sentimento negativo do cliente em um positivo à medida que as chamadas progridem.
- Melhorar a organização dos dados: rotule e classifique as chamadas com base em [categorias personalizadas](#) (incluindo palavras-chave e frases, sentimentos, tempo de conversa e interrupções).
- Resumir os aspectos importantes de uma chamada usando IA generativa: use o [resumo generativo da chamada](#) para receber um resumo conciso da transcrição, que inclui os principais componentes, como problemas, itens de ação e resultados discutidos na chamada.

Transcrições em tempo real:

- Reduzir a tensão em tempo real: configure [alertas em tempo real](#) para as principais frases, como um cliente dizendo “falar com um gerente”, para sinalizar quando as chamadas começam a ficar mais tensas. Você pode criar alertas em tempo real usando correspondências de categorias em tempo real.
- Melhorar o manuseio dos dados dos clientes: use a [identificação de PII](#) ou a [edição de PII](#) na saída da transcrição para ajudar a proteger a privacidade do cliente.
- Identificar palavras-chave e frases personalizadas: use [categorias personalizadas](#) para sinalizar palavras-chave específicas em uma chamada.

- Identificar problemas automaticamente: use a [detecção de problemas](#) automática para receber um resumo sucinto de todos os problemas identificados em uma chamada.
- Avalie a eficácia da equipe na criação de uma experiência positiva para o cliente: use a [análise de sentimentos](#) para avaliar se os atendentes conseguem transformar um sentimento negativo do cliente em positivo à medida que as chamadas progridem.
- Configurar a assistência ao atendente: use os insights de sua escolha para fornecer assistência proativa aos atendentes na resolução de chamadas de clientes. Consulte [Live Call Analytics and agent assist for your contact center with Amazon language AI services](#) para obter mais informações.

Para comparar os recursos disponíveis com o Call Analytics com Amazon Transcribe os do Amazon Transcribe Medical, consulte a [tabela de recursos](#).

Para começar, consulte [Iniciar uma transcrição de análise pós-chamada](#) e [Iniciar uma transcrição do Call Analytics em tempo real](#). A saída do Call Analytics é semelhante à de um trabalho de transcrição padrão, mas contém dados analíticos adicionais. Para ver um exemplo de saída, consulte [Saída da análise pós-chamada](#) e [Saída de análise de chamadas em tempo real](#).

Considerações e informações adicionais

Antes de usar o Call Analytics, observe que:

- O Call Analytics só é compatível com áudio de dois canais, em que um atendente está presente em um canal e um cliente está presente em um segundo canal.
- O [Fila de Job](#) está sempre habilitado para trabalhos de análise pós-chamada, então há um limite de cem trabalhos simultâneos de análise de chamadas. Se você quiser solicitar um aumento de cota, consulte [AWS service quotas](#).
- Os arquivos de entrada para trabalhos de análise pós-chamada não podem ter mais de 500 MB e devem ter menos de 4 horas. Observe que o limite de tamanho do arquivo pode ser menor para determinados formatos de arquivo de áudio compactado e não WAV.
- Se estiver usando categorias, você deverá criar todas as categorias desejadas antes de iniciar uma transcrição do Call Analytics. Nenhuma categoria nova pode ser aplicada às transcrições existentes. Para saber como criar uma nova categoria, consulte [Criar categorias para transcrições pós-chamada](#) e [Criar categorias para transcrições em tempo real](#).
- Algumas cotas do Call Analytics são diferentes das do Amazon Transcribe setor Amazon Transcribe médico; consulte a [Referência AWS Geral](#) para obter detalhes.

i Mergulhe mais fundo com o Blog AWS de Machine Learning

Para saber mais sobre as opções do Call Analytics, consulte:

- [Post Call Analytics for your contact center with Amazon language AI services](#)
- [Live Call Analytics and agent assist for your contact center with Amazon language AI services](#)

Para ver exemplos de resultados e recursos do Call Analytics, veja nossa [GitHubdemonstração](#). Também oferecemos um aplicativo de [documento JSON para Word](#) para converter sua transcrição em um easy-to-read formato.

Disponibilidade e cotas da região

O Call Analytics é compatível com o seguinte Regiões da AWS:

Region	Tipo de transcrição
ap-northeast-1 (Tóquio)	post-call, real-time
ap-northeast-2 (Seul)	post-call, real-time
ap-south-1 (Mumbai)	post-call
ap-southeast-1 (Singapura)	post-call
ap-southeast-2 (Sydney)	post-call, real-time
ca-central-1: (Canadá, Central)	post-call, real-time
eu-central-1 (Frankfurt)	post-call, real-time
eu-west-2 (Londres)	post-call, real-time
us-east-1 (N. Virginia)	post-call, real-time
us-west-2 (Oregon)	post-call, real-time

Observe que o suporte de região é diferente para o [Amazon Transcribe](#), o [Amazon Transcribe Medical](#) e o Call Analytics.

Para obter os endpoints de cada região com suporte, consulte [Service endpoints](#) na Referência geral da AWS .

Para obter uma lista de cotas referentes às suas transcrições, consulte as [Service quotas](#) na Referência geral da AWS . Algumas cotas podem ser alteradas mediante solicitação. Se a coluna Ajustável contiver “Sim”, você poderá solicitar um aumento. Para fazer isso, selecione o link fornecido.

Análise pós-chamada

O Call Analytics fornece análises pós-chamada, que são úteis para monitorar as tendências do atendimento ao cliente.

As transcrições pós-chamada oferecem os seguintes insights:

- [Características da chamada](#), incluindo tempo de conversa, tempo sem conversa, volume do locutor, interrupções, velocidade da conversa, problemas, resultados e itens de ação
- [Resumo generativo da chamada](#), que cria um resumo conciso de toda a chamada
- [Categorização personalizada](#) com regras que você pode usar para aprimorar palavras-chave e critérios específicos
- [Edição de PII](#) da transcrição de texto e do arquivo de áudio.
- [Sentimento do locutor](#) para cada pessoa em vários momentos de uma chamada.

Insights pós-chamada

Essa seção detalha os insights disponíveis para transcrições de análises pós-chamada.

Características da chamada

O recurso de características da chamada avalia a qualidade das interações entre atendente e cliente usando estes critérios:

- **Interrupção:** mede se e quando um participante interrompe o outro participante no meio da frase. Interrupções frequentes podem estar associadas a grosseria ou raiva e podem se correlacionar com sentimentos negativos de um ou de ambos os participantes.

- **Volume:** mede o volume em que cada participante está falando. Use essa métrica para ver se o chamador ou o atendente está falando alto ou gritando, o que geralmente indica irritação. Essa métrica é representada como um valor normalizado (nível de fala por segundo em determinado segmento) em uma escala de 0 a 100, onde um valor mais alto indica uma voz mais alta.
- **Tempo sem conversa:** mede períodos que não contêm fala. Use essa métrica para ver se há longos períodos de silêncio, como um atendente que mantém um cliente em espera por um período excessivo de tempo.
- **Velocidade da conversa:** mede a velocidade com que os dois participantes estão falando. A compreensão pode ser afetada se um participante falar muito rápido. Essa métrica é medida em palavras por minuto.
- **Tempo de conversa:** mede o tempo de fala (em milissegundos) de cada participante durante a chamada. Use essa métrica para ajudar a identificar se um participante está dominando a chamada ou se o diálogo está equilibrado.
- **Problemas, resultados e itens de ação:** identifica problemas, resultados e itens de ação da transcrição da chamada.

Veja a seguir um [exemplo de saída](#).

Resumo generativo de chamada

O resumo generativo da chamada cria um resumo conciso de toda a chamada, capturando os principais componentes, como o motivo da chamada, as etapas tomadas para resolver o problema e as próximas etapas.

Com o resumo generativo de chamada, é possível:

- Reduzir a necessidade de anotações manuais durante e após as chamadas.
- Melhorar a eficiência dos atendentes, pois eles podem passar mais tempo conversando com as pessoas que aguardam na fila do que no trabalho pós-chamada.
- Acelerar as avaliações do supervisor, pois os resumos das chamadas são muito mais rápidos de revisar do que as transcrições inteiras.

Para usar o resumo generativo de chamada com um trabalho de análise pós-chamada, consulte [Enabling generative call summarization](#). Consulte um exemplo de saída em [Generative call summarization output example](#). O resumo generativo de chamada tem um preço separado (consulte a [página de preços](#)).

Note

No momento, o resumo generativo de chamada está disponível nas regiões `us-east-1` e `us-west-2`. Esse recurso é compatível com os seguintes dialetos do idioma inglês: australiano (`en-AU`), britânico (`en-GB`), indiano (`en-IN`), irlandês (`en-IE`), escocês (`en-AB`), EUA (`en-US`) e galês (`en-WL`).

Categorização personalizada

Use a categorização de chamadas para sinalizar palavras-chave, frases, sentimentos ou ações em uma chamada. Nossas opções de categorização podem ajudar na triagem de aumento da tensão, como chamadas com sentimentos negativos com muitas interrupções, ou na organização de chamadas em categorias específicas, como departamentos da empresa.

Os critérios que você pode adicionar a uma categoria incluem:

- Tempo sem conversa: períodos em que nem o cliente nem o atendente estão falando.
- Interrupções: quando o cliente ou o atendente interrompe a outra pessoa.
- Sentimento do cliente ou do atendente: como o cliente ou o atendente estão se sentindo durante um período específico. Se pelo menos 50% da conversa girar (back-and-forth entre dois palestrantes) em um período de tempo especificado corresponder ao sentimento especificado, Amazon Transcribe considere o sentimento uma correspondência.
- Palavras-chave ou frases: faz a correspondência de parte da transcrição com base em uma frase exata. Por exemplo, se você definir um filtro para a frase “Quero falar com o gerente”, o Amazon Transcribe filtrará exatamente essa frase.

Você também pode sinalizar o inverso dos critérios anteriores (tempo de conversa, falta de interrupções, ausência de um sentimento e falta de uma frase específica).

Veja a seguir um [exemplo de saída](#).

Para obter mais informações sobre categorias ou saber como criar uma categoria, consulte [Criar categorias para transcrições pós-chamada](#).

Você pode criar quantas categorias quiser para abranger uma variedade de cenários diferentes. Para cada categoria criada, você deve criar entre 1 e 20 regras. Cada regra é baseada em um destes quatro critérios: interrupções, palavras-chave, tempo sem conversa ou sentimento. Para obter mais detalhes sobre o uso desses critérios com a operação [CreateCallAnalyticsCategory](#), consulte a seção [Critérios de regras para categorias de análise pós-chamada](#).

Se o conteúdo da mídia corresponder a todas as regras que você especificou em determinada categoria, o Amazon Transcribe rotulará a saída com essa categoria. Consulte a [saída de categorização de chamadas](#) para ver um exemplo de correspondência de categoria na saída JSON.

Veja a seguir alguns exemplos do que você pode fazer com categorias personalizadas:

- Isolar chamadas com características específicas, como chamadas que terminam com um sentimento negativo do cliente.
- Identificar tendências nos problemas dos clientes sinalizando e rastreando conjuntos específicos de palavras-chave.
- Monitorar a conformidade; por exemplo, um atendente que está falando (ou omitindo) uma frase específica durante os primeiros segundos de uma chamada.
- Obter insights sobre a experiência do cliente sinalizando chamadas em que há muitas interrupções por parte dos atendentes e sentimentos negativos do cliente
- Comparar várias categorias para avaliar as correlações; por exemplo, analisar se uma frase de boas-vindas usada pelo atendente se correlaciona com o sentimento positivo do cliente.

Categorias pós-chamada versus categorias em tempo real

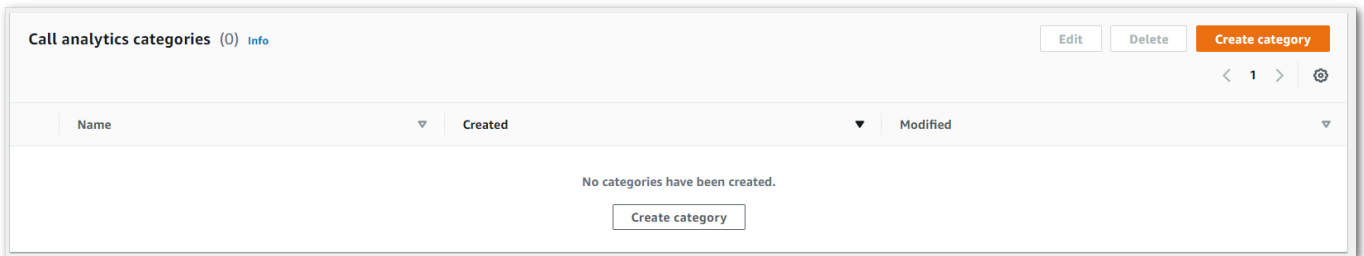
Ao criar uma categoria, você pode especificar se deseja que ela seja criada como uma categoria de análise pós-chamada (POST_CALL) ou como uma categoria de análise de chamadas em tempo real (REAL_TIME). Se não for especificada uma opção, a categoria será criada como uma categoria de pós-chamada por padrão. As correspondências de categoria de análise pós-chamada estão disponíveis na saída após a conclusão da transcrição da análise pós-chamada.

Para criar uma categoria para análise pós-chamada, você pode usar o AWS Management Console, a AWS CLI ou os SDKs da AWS . Veja os seguintes exemplos:

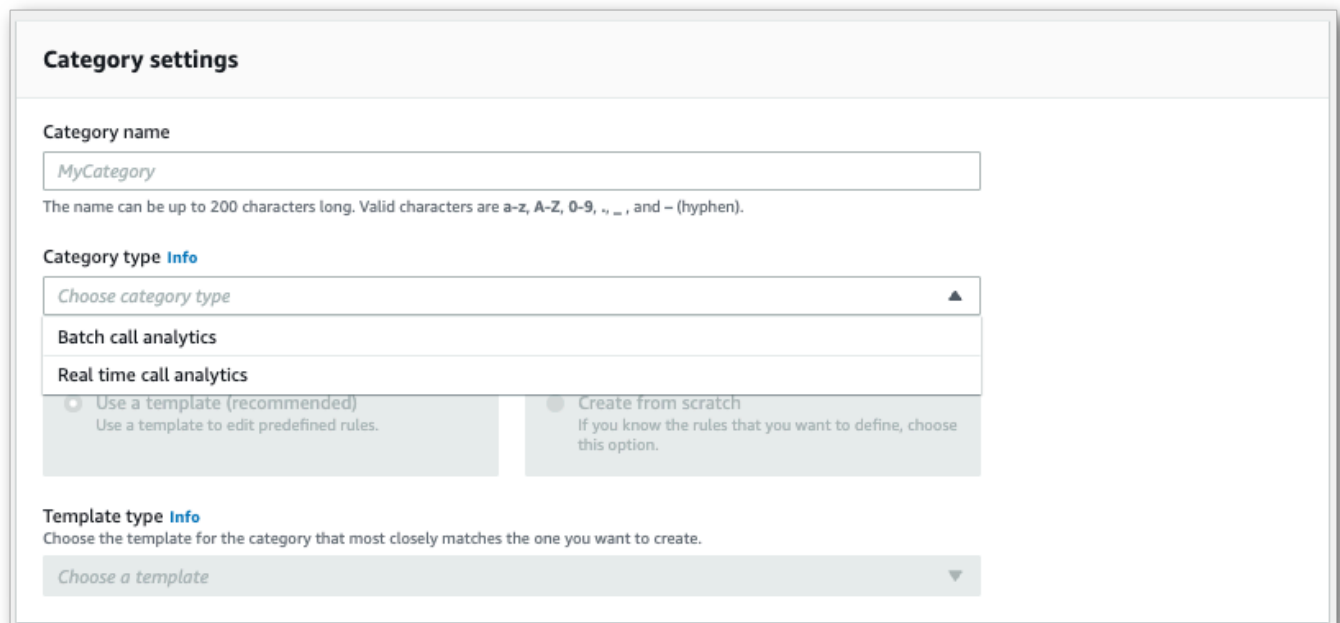
AWS Management Console

1. No painel de navegação, em Amazon Transcribe, escolha Análise de Amazon Transcribe chamadas.

- Escolha Categorias de análise de chamadas, que levará você à página Categorias de análise de chamadas. Selecione Criar categoria.



- Agora você está na página Criar categoria. Insira um nome para a categoria e escolha “Análise de chamadas em lote” no menu suspenso Tipo de categoria.



- Você pode escolher um modelo para criar a categoria ou criar uma do zero.

Se estiver usando um modelo: selecione Usar um modelo (recomendado), escolha o modelo desejado e selecione Criar categoria.

Category settings

Category name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, ., _ , and - (hyphen).

Category type [Info](#)

Category creation method [Info](#)

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Template type [Info](#)

Choose the template for the category that most closely matches the one you want to create.

- Non-talk time exceeds 5 minutes for the whole call
- Customer sentiment is negative for the last 5 minutes of the call
- Agent spoke over the customer more than 15 seconds for the entire call

ALL THE GUIS CONDITIONS MUST BE MET FOR A REPRESENTATIVE TO BE ASSIGNED TO THE CATEGORY.

5. Se estiver criando uma categoria personalizada: selecione Criar do zero.

Create category [Info](#)

Category settings

Category name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, ., _ , and - (hyphen).

Category creation method [Info](#)

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Rules

All the rule conditions must be met for a transcription job to be classified in this category.

▼ Rule 1 Delete rule

Rule type [Info](#)
Choose the rule that you want to define.

[Add rule](#)

You can add up to 19 more rules.

6. Adicione regras à categoria usando o menu suspenso. Você pode adicionar até vinte regras por categoria.

The screenshot shows the 'Rules' configuration page in Amazon Transcribe. At the top, it states: 'All the rule conditions must be met for a transcription job to be classified in this category.' Below this, there is a section for 'Rule 1' with a 'Delete rule' button. The rule description is: 'When no word has been spoken for more than 5 minute(s) during the entire call.' Underneath, the 'Rule type' is set to 'Non-talk time', with an 'Info' link. A dropdown menu is open, showing four options: 'Non-talk time' (selected), 'Interruption time', 'Transcript content match', and 'Transcript sentiment match'. Each option has a brief description of when the rule is triggered. At the bottom of the dropdown is an 'Add rule' button. Below the dropdown, it says 'You can add up to 19 more rules.'

7. Veja um exemplo de uma categoria com duas regras: um atendente que interrompe um cliente por mais de 15 segundos durante a chamada e um sentimento negativo do cliente ou do atendente nos últimos 2 minutos da chamada.

Rules

All the rule conditions must be met for a transcription job to be classified in this category.

▼ **Rule 1** Delete rule

When the duration of the interruption was more than 15 second(s) during the entire call when the speaker was agent.

Rule type [Info](#)
Choose the rule that you want to define.

Interruption time ▼

Logic [Info](#)
Define the conditions that must be met.

When the duration of the interruption was more than ▼

during the ▼

when the speaker was ▼

AND

▼ **Rule 2** Delete rule

When the sentiment is negative during the last 2 minute(s) when the speaker was either.

Rule type [Info](#)
Choose the rule that you want to define.

Transcript sentiment match ▼

Logic [Info](#)
Define the conditions that must be met.

When the sentiment is ▼

during the ▼ ▼

when the speaker was ▼

You can add up to 18 more rules.

8. Quando terminar de adicionar regras à sua categoria, escolha Criar categoria.

AWS CLI

Este exemplo usa o comando [create-call-analytics-category](#). Para obter mais informações, consulte [CreateCallAnalyticsCategory](#), [CategoryProperties](#) e [Rule](#).

O seguinte exemplo cria uma categoria com as regras:

- O cliente foi interrompido nos primeiros 60.000 milissegundos. A duração dessas interrupções durou pelo menos 10.000 milissegundos.
- Houve um período de silêncio que durou pelo menos 20.000 milissegundos entre 10% da chamada e 80% da chamada.
- O atendente teve um sentimento negativo em algum momento da chamada.
- As palavras “bem-vindo” ou “olá” não foram usadas nos primeiros 10.000 da chamada.

Este exemplo usa o comando [create-call-analytics-category](#) e um corpo de solicitação que adiciona várias regras à categoria.

```
aws transcribe create-call-analytics-category \  
--cli-input-json file://filepath/my-first-analytics-category.json
```

O arquivo my-first-analytics-category.json contém o corpo da solicitação a seguir.

```
{  
  "CategoryName": "my-new-category",  
  "InputType": "POST_CALL",  
  "Rules": [  
    {  
      "InterruptionFilter": {  
        "AbsoluteTimeRange": {  
          "First": 60000  
        },  
        "Negate": false,  
        "ParticipantRole": "CUSTOMER",  
        "Threshold": 10000  
      }  
    },  
    {  
      "NonTalkTimeFilter": {  
        "Negate": false,  
        "RelativeTimeRange": {  
          "EndPercentage": 80,  
          "StartPercentage": 10  
        },  
        "Threshold": 20000  
      }  
    },  
    {
```

```

    "SentimentFilter": {
      "ParticipantRole": "AGENT",
      "Sentiments": [
        "NEGATIVE"
      ]
    },
    {
      "TranscriptFilter": {
        "Negate": true,
        "AbsoluteTimeRange": {
          "First": 10000
        },
        "Targets": [
          "welcome",
          "hello"
        ],
        "TranscriptFilterType": "EXACT"
      }
    }
  ]
}

```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para criar uma categoria usando os `Rules` argumentos `CategoryName` e para o método [create_call_analytics_category](#). Para obter mais informações, consulte [CreateCallAnalyticsCategory](#), [CategoryProperties](#) e [Rule](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

O seguinte exemplo cria uma categoria com as regras:

- O cliente foi interrompido nos primeiros 60.000 milissegundos. A duração dessas interrupções durou pelo menos 10.000 milissegundos.
- Houve um período de silêncio que durou pelo menos 20.000 milissegundos entre 10% da chamada e 80% da chamada.
- O atendente teve um sentimento negativo em algum momento da chamada.
- As palavras “bem-vindo” ou “olá” não foram usadas nos primeiros 10.000 da chamada.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
category_name = "my-new-category"
transcribe.create_call_analytics_category(
    CategoryName = category_name,
    InputType = POST_CALL,
    Rules = [
        {
            'InterruptionFilter': {
                'AbsoluteTimeRange': {
                    'First': 60000
                },
                'Negate': False,
                'ParticipantRole': 'CUSTOMER',
                'Threshold': 10000
            }
        },
        {
            'NonTalkTimeFilter': {
                'Negate': False,
                'RelativeTimeRange': {
                    'EndPercentage': 80,
                    'StartPercentage': 10
                },
                'Threshold': 20000
            }
        },
        {
            'SentimentFilter': {
                'ParticipantRole': 'AGENT',
                'Sentiments': [
                    'NEGATIVE'
                ]
            }
        },
        {
            'TranscriptFilter': {
                'Negate': True,
                'AbsoluteTimeRange': {
                    'First': 10000
                },
            },
        }
    ]
)
```

```

        'Targets': [
            'welcome',
            'hello'
        ],
        'TranscriptFilterType': 'EXACT'
    }
}
]
)

result = transcribe.get_call_analytics_category(CategoryName = category_name)
print(result)

```

Crítérios de regras para categorias de análise pós-chamada

Esta seção descreve os tipos de regras personalizadas POST_CALL que podem ser criadas usando a operação de API [CreateCallAnalyticsCategory](#).

Correspondência de interrupção

As regras que usam interrupções (tipo de dados [InterruptionFilter](#)) foram projetadas para corresponder a:

- Instâncias em que um atendente interrompe um cliente
- Situações em que um cliente interrompe um atendente
- Interrupção de qualquer um dos participantes
- Falta de interrupções

Veja abaixo um exemplo dos parâmetros disponíveis com [InterruptionFilter](#):

```

"InterruptionFilter": {
  "AbsoluteTimeRange": {
    Specify the time frame, in milliseconds, when the match should occur
  },
  "RelativeTimeRange": {
    Specify the time frame, in percentage, when the match should occur
  },
  "Negate": Specify if you want to match the presence or absence of interruptions,
  "ParticipantRole": Specify if you want to match speech from the agent, the customer, or both,

```

```
"Threshold": Specify a threshold for the amount of time, in seconds, interruptions occurred during the call
},
```

Consulte [CreateCallAnalyticsCategory](#) e [InterruptionFilter](#) para obter mais informações sobre esses parâmetros e os valores válidos associados a cada um.

Correspondência de palavras-chave

As regras que usam palavras-chave (tipo de dados [TranscriptFilter](#)) foram criadas para corresponder a:

- Palavras ou frases personalizadas faladas pelo atendente, pelo cliente ou por ambos
- Palavras ou frases personalizadas não ditas pelo atendente, pelo cliente ou por ambos
- Palavras ou frases personalizadas que ocorrem em um período específico

Veja abaixo um exemplo dos parâmetros disponíveis com [TranscriptFilter](#):

```
"TranscriptFilter": {
  "AbsoluteTimeRange": {
    Specify the time frame, in milliseconds, when the match should occur
  },
  "RelativeTimeRange": {
    Specify the time frame, in percentage, when the match should occur
  },
  "Negate": Specify if you want to match the presence or absence of your custom keywords,
  "ParticipantRole": Specify if you want to match speech from the agent, the customer, or both,
  "Targets": [ The custom words and phrases you want to match ],
  "TranscriptFilterType": Use this parameter to specify an exact match for the specified targets
}
```

Consulte [CreateCallAnalyticsCategory](#) e [TranscriptFilter](#) para obter mais informações sobre esses parâmetros e os valores válidos associados a cada um.

Correspondência de tempo sem conversa

As regras que usam tempo sem conversa (tipo de dados [NonTalkTimeFilter](#)) foram projetadas para corresponder a:

- Presença de silêncio em períodos específicos durante a chamada
- Presença de fala em períodos específicos durante a chamada

Veja abaixo um exemplo dos parâmetros disponíveis com [NonTalkTimeFilter](#):

```
"NonTalkTimeFilter": {
  "AbsoluteTimeRange": {
    Specify the time frame, in milliseconds, when the match should occur
  },
  "RelativeTimeRange": {
    Specify the time frame, in percentage, when the match should occur
  },
  "Negate": Specify if you want to match the presence or absence of speech,
  "Threshold": Specify a threshold for the amount of time, in seconds, silence (or speech) occurred during the call
},
```

Consulte [CreateCallAnalyticsCategory](#) e [NonTalkTimeFilter](#) para obter mais informações sobre esses parâmetros e os valores válidos associados a cada um.

Correspondência de sentimentos

As regras que usam sentimento (tipo de dados [SentimentFilter](#)) foram criadas para corresponder a:

- Presença ou ausência de um sentimento positivo expresso pelo cliente, pelo atendente ou por ambos em pontos específicos da chamada
- Presença ou ausência de um sentimento negativo expresso pelo cliente, pelo atendente ou por ambos em pontos específicos da chamada
- Presença ou ausência de um sentimento neutro expresso pelo cliente, pelo atendente ou por ambos em pontos específicos da chamada
- Presença ou ausência de um sentimento misto expresso pelo cliente, pelo atendente ou por ambos em pontos específicos da chamada

Veja abaixo um exemplo dos parâmetros disponíveis com [SentimentFilter](#):

```
"SentimentFilter": {
  "AbsoluteTimeRange": {
```



```
Specify the time frame, in milliseconds, when the match should occur
},
"RelativeTimeRange": {
Specify the time frame, in percentage, when the match should occur
},
"Negate": Specify if you want to match the presence or absence of your chosen sentiment,
"ParticipantRole": Specify if you want to match speech from the agent, the customer, or both,
"Sentiments": [ The sentiments you want to match ]
},
```

Consulte [CreateCallAnalyticsCategory](#) e [SentimentFilter](#) para obter mais informações sobre esses parâmetros e os valores válidos associados a cada um.

Iniciar uma transcrição de análise pós-chamada

Antes de iniciar uma transcrição de análise pós-chamada, você deve criar todas as [categorias](#) que deseja Amazon Transcribe combinar em seu áudio.

Note

As transcrições do Call Analytics não podem ser combinadas retroativamente com novas categorias. Somente as categorias que você cria antes de iniciar uma transcrição do Call Analytics podem ser aplicadas a essa saída de transcrição.

Se você criou uma ou mais categorias e seu áudio corresponde a todas as regras em pelo menos uma de suas categorias, o Amazon Transcribe sinaliza a saída com a categoria correspondente. Se você optar por não usar categorias ou se o áudio não corresponder às regras especificadas nas categorias, sua transcrição não será sinalizada.

Para iniciar uma transcrição de análise pós-chamada, você pode usar o AWS Management Console, a AWS CLI ou os SDKs da AWS ; veja exemplos a seguir:

AWS Management Console

Siga o procedimento a seguir para iniciar um trabalho de análise pós-chamada. As chamadas que correspondem a todas as características definidas por uma categoria são rotuladas com essa categoria.

1. No painel de navegação, em Análise de Amazon Transcribe chamadas, escolha Trabalhos de análise de chamadas.
2. Escolha Criar trabalho.

Configure job - *optional* [Info](#)

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)

Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)

Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)

A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Summarization

Generative call summarization [Info](#)

Generative call summarization provides a summary of the transcript, including important components of the conversation.

Categories

Create categories to classify calls. For example, you can create a category for all cancellation requests. When you run an analytics job, Amazon Transcribe applies that category to all calls that request cancellation.

Call analytics categories (1) [Info](#)

< 1 > 

	Name	Type	Created	Modified
<input type="radio"/>	CatchNegativeSentiment	POST_CALL	February 17 2023, 10:43 (UTC-08:00)	February 17 2023, 10:43 (UTC-08:00)

If the above categories aren't relevant to your use case, you can create a new category. [Create a new category.](#) [↗](#)

Cancel

Previous

Create job

3. Na página Especificar os detalhes da tarefa, forneça informações sobre o trabalho de análise de chamadas, incluindo a localização dos dados de entrada.

Specify job details [Info](#)

Job settings

Name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, . (period), _ (underscore), and - (hyphen).

Model type [Info](#)
Choose the type of model to use for the transcription job.

General model
To use a model that is not specialized for a particular use case, choose this option. Configuration options vary between languages.

Custom language model
To use a model that you trained for your specific use case, choose this option. This model has fewer configuration options than the general model.

Language settings
You can transcribe your audio file in a language that you specify or have Amazon Transcribe identify and transcribe it in the predominant language.

Specific language [Info](#)
If you know the language spoken in your source audio, choose this option to get the most accurate results. The options available for additional processing vary between languages.

Automatic language identification [Info](#)
If you don't know the language spoken in your audio files, choose this option. You have access to fewer options for additional processing than if you choose **Specific language**.

Language
Choose the language of the input audio.

Input data [Info](#)

Input file location on S3
Choose an input audio or video file in Amazon S3.

Valid file formats: MP3, MP4, WAV, FLAC, AMR, OGG, and WebM.

Agent audio channel identification [Info](#)
Choose the channel that has the speech from the agent. The other channel is used for the customer's speech.

Especifique a Amazon S3 localização desejada dos dados de saída e qual IAM função usar.

Output data

Output data location type info [Info](#)

Service-managed S3 bucket
The output will be removed after 90 days when the job expires.

Customer specified S3 bucket
The output will not be removed from bucket even after the job expires.

Access permissions

IAM role [Info](#)

Use an existing IAM role

Create an IAM role
By choosing **Create job** you are authorizing creation of this role.

Permissions to access
Your role has access to these resources. The KMS key permission is used only if your input bucket is encrypted

Input S3 bucket and KMS decrypt permission to input bucket

Any S3 bucket and any KMS keys

Role name
Roles are prefixed with "AmazonTranscribeServiceRoleFullAccess-". Your newly created role has full access to the S3 bucket and KMS key for your account.

The name can be up to 64 characters long

▼ **Role permissions details**

Your new role has these permissions to give Amazon Transcribe access to the resources that you've specified.

Service	Access level	Resource
S3	List, Read, Write	All resources
Key Management Service	GenerateDataKey, Decrypt	All resources

Cancel **Next**

4. Escolha Próximo.
5. Em Configurar tarefa, ative os recursos opcionais que você deseja incluir no trabalho de análise de chamadas. Se você criou categorias anteriormente, elas aparecem no painel Categorias e são aplicadas automaticamente ao trabalho de análise de chamadas.

Configure job - *optional* [Info](#)

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)

Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)

Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)

A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Summarization

Generative call summarization [Info](#)

Generative call summarization provides a summary of the transcript, including important components of the conversation.


Categories

Create categories to classify calls. For example, you can create a category for all cancellation requests. When you run an analytics job, Amazon Transcribe applies that category to all calls that request cancellation.

Call analytics categories (1) [Info](#)

< 1 > 

	Name	Type	Created	Modified
<input type="radio"/>	CatchNegativeSentiment	POST_CALL	February 17 2023, 10:43 (UTC-08:00)	February 17 2023, 10:43 (UTC-08:00)

If the above categories aren't relevant to your use case, you can create a new category. [Create a new category.](#) 

6. Escolha Criar trabalho.

AWS CLI

Este exemplo usa o comando [start-call-analytics-job](#) e o parâmetro `channel-definitions`. Para obter mais informações, consulte [StartCallAnalyticsJob](#) e [ChannelDefinition](#).

```
aws transcribe start-call-analytics-job \  
--region us-west-2 \  
--call-analytics-job-name my-first-call-analytics-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-location s3://DOC-EXAMPLE-BUCKET/my-output-files/ \  
--data-access-role-arn arn:aws:iam::111122223333:role/ExampleRole \  
--channel-definitions ChannelId=0,ParticipantRole=AGENT \  
ChannelId=1,ParticipantRole=CUSTOMER
```

Veja está outro exemplo que usa o comando [start-call-analytics-job](#) e um corpo de solicitação que habilita o Call Analytics para esse trabalho.

```
aws transcribe start-call-analytics-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-call-analytics-job.json
```

O arquivo `my-call-analytics-job.json` contém o corpo da solicitação a seguir.

```
{  
  "CallAnalyticsJobName": "my-first-call-analytics-job",  
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputLocation": "s3://DOC-EXAMPLE-BUCKET/my-output-files/",  
  "ChannelDefinitions": [  
    {  
      "ChannelId": 0,  
      "ParticipantRole": "AGENT"  
    },  
    {  
      "ChannelId": 1,  
      "ParticipantRole": "CUSTOMER"  
    }  
  ]  
}
```



```
]
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para iniciar um trabalho do Call Analytics usando o método [start_call_analytics_job](#). Para obter mais informações, consulte [StartCallAnalyticsJob](#) e [ChannelDefinition](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-call-analytics-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
output_location = "s3://DOC-EXAMPLE-BUCKET/my-output-files/"
data_access_role = "arn:aws:iam::111122223333:role/ExampleRole"
transcribe.start_call_analytics_job(
    CallAnalyticsJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    DataAccessRoleArn = data_access_role,
    OutputLocation = output_location,
    ChannelDefinitions = [
        {
            'ChannelId': 0,
            'ParticipantRole': 'AGENT'
        },
        {
            'ChannelId': 1,
            'ParticipantRole': 'CUSTOMER'
        }
    ]
)

while True:
    status = transcribe.get_call_analytics_job(CallAnalyticsJobName = job_name)
    if status['CallAnalyticsJob']['CallAnalyticsJobStatus'] in ['COMPLETED', 'FAILED']:
```

```
break
print("Not ready yet...")
time.sleep(5)
print(status)
```

Saída da análise pós-chamada

As transcrições da análise pós-chamada são exibidas em turn-by-turn formato por segmento. Elas incluem categorização de chamadas, características da chamada (pontuações de volume, interrupções, tempo sem conversa, velocidade da conversa), resumo da chamada (problemas, resultados e itens de ação), edição e sentimento. Além disso, um resumo das características da conversa é fornecido no fim da transcrição.

Para aumentar a precisão e personalizar ainda mais as transcrições de acordo com seu caso de uso, como incluir termos específicos do setor, adicione [vocabulários personalizados](#) ou [modelos de idioma personalizados](#) à solicitação de análise de chamadas. Para mascarar, remover ou marcar palavras que você não deseja nos resultados da transcrição, como palavrões, adicione [filtragem de vocabulário](#). Se não tiver certeza do código do idioma a ser passado para o arquivo de mídia, você poderá habilitar a [identificação de idioma em lote](#) para identificar automaticamente o idioma no arquivo de mídia.

As seções a seguir mostram exemplos de saída JSON em um nível de insight. Para obter a saída compilada, consulte [Saída de análise pós-chamada compilada](#).

Categorização de chamadas

Veja a aparência de uma correspondência de categoria na saída da transcrição. Este exemplo mostra que o áudio do carimbo de data e hora de 40.040 milissegundos para o carimbo de data e hora de 42.460 milissegundos corresponde à categoria “resolução positiva”. Nesse caso, a categoria personalizada de “resolução positiva” exigia um sentimento positivo nos últimos segundos da fala.

```
"Categories": {
  "MatchedDetails": {
    "positive-resolution": {
      "PointsOfInterest": [
        {
          "BeginOffsetMillis": 40040,
          "EndOffsetMillis": 42460
        }
      ]
    }
  }
}
```

```
    }
  },
  "MatchedCategories": [
    " positive-resolution"
  ]
},
```

Características da chamada

Veja como são as características da chamada na saída de transcrição. Observe que as pontuações de intensidade são fornecidas para cada turno de conversa, enquanto todas as outras características são fornecidas no final da transcrição.

```
"LoudnessScores": [
  87.54,
  88.74,
  90.16,
  86.36,
  85.56,
  85.52,
  81.79,
  87.74,
  89.82
],
...

"ConversationCharacteristics": {
  "NonTalkTime": {
    "Instances": [],
    "TotalTimeMillis": 0
  },
  "Interruptions": {
    "TotalCount": 2,
    "TotalTimeMillis": 10700,
    "InterruptionsByInterrupter": {
      "AGENT": [
        {
          "BeginOffsetMillis": 26040,
          "DurationMillis": 5510,
          "EndOffsetMillis": 31550
        }
      ]
    }
  },
  ],
```

```

    "CUSTOMER": [
      {
        "BeginOffsetMillis": 770,
        "DurationMillis": 5190,
        "EndOffsetMillis": 5960
      }
    ]
  },
  "TotalConversationDurationMillis": 42460,
  ...,
  "TalkSpeed": {
    "DetailsByParticipant": {
      "AGENT": {
        "AverageWordsPerMinute": 150
      },
      "CUSTOMER": {
        "AverageWordsPerMinute": 167
      }
    }
  },
  "TalkTime": {
    "DetailsByParticipant": {
      "AGENT": {
        "TotalTimeMillis": 32750
      },
      "CUSTOMER": {
        "TotalTimeMillis": 18010
      }
    },
    "TotalTimeMillis": 50760
  }
},

```

Problemas, itens de ação e próximas etapas

- No exemplo a seguir, os problemas são identificados como começando no caractere 7 e terminando no caractere 51, que se refere a esta seção do texto: “Eu gostaria de cancelar minha assinatura de receitas”.

```
"Content": "Well, I would like to cancel my recipe subscription.",
```

```
"IssuesDetected": [
  {
    "CharacterOffsets": {
      "Begin": 7,
      "End": 51
    }
  }
],
```

- No exemplo a seguir, os resultados são identificados como começando no caractere 12 e terminando no caractere 78, que se refere a esta seção do texto: “Eu fiz todas as alterações em sua conta e agora esse desconto está aplicado”.

```
"Content": "Wonderful. I made all changes to your account and now this discount is applied, please check.",
"OutcomesDetected": [
  {
    "CharacterOffsets": {
      "Begin": 12,
      "End": 78
    }
  }
],
```

- No exemplo a seguir, os itens de ação são identificados como começando no caractere 0 e terminando no caractere 103, que se refere a esta seção do texto: “Enviarei um e-mail com todos os detalhes para você hoje e ligarei de volta na próxima semana para fazer um acompanhamento”.

```
"Content": "I will send an email with all the details to you today, and I will call you back next week to follow up. Have a wonderful evening.",
"ActionItemsDetected": [
  {
    "CharacterOffsets": {
      "Begin": 0,
      "End": 103
    }
  }
],
```

Resumo generativo de chamada

Veja como é o resumo generativo de chamada na saída de transcrição:

```
"ContactSummary": {
  "AutoGenerated": {
    "OverallSummary": {
      "Content": "A customer wanted to check to see if we had a bag allowance. We
told them that we didn't have it, but we could add the bag from Canada to Calgary and
then do the one coming back as well."
    }
  }
}
```

O trabalho de análise será concluído sem geração de resumo nos seguintes casos:

- Conteúdo insuficiente da conversa: a conversa deve incluir pelo menos uma vez do agente e do cliente. Quando o conteúdo da conversa for insuficiente, o serviço retornará o código de erro `INSUFFICIENT_CONVERSATION_CONTENT`.
- Proteções de segurança: a conversa deve atender às barreiras de segurança estabelecidas para garantir que o resumo apropriado seja gerado. Quando essas grades de proteção não forem atendidas, o serviço retornará o código de erro `FAILED_SAFETY_GUIDELINES`.

O código de erro pode ser encontrado na `Skipped` seção dentro `AnalyticsJobDetails` da saída. Você também pode encontrar o motivo do erro `CallAnalyticsJobDetails` na Resposta da [GetCallAnalyticsJobAPI](#).

Exemplo de saída de erro

```
{
  "JobStatus": "COMPLETED",
  "AnalyticsJobDetails": {
    "Skipped": [
      {
        "Feature": "GENERATIVE_SUMMARIZATION",
        "ReasonCode": "INSUFFICIENT_CONVERSATION_CONTENT",
        "Message": "The conversation needs to have at least one turn from both
the participants to generate summary"
      }
    ]
  }
}
```

```

    },
    "LanguageCode": "en-US",
    "AccountId": "*****",
    "JobName": "Test2-copy",
    ...
}

```

Análise de sentimentos

Veja como é a análise de sentimentos na saída de transcrição.

- Valores qualitativos turn-by-turn do sentimento:

```

"Content": "That's very sad to hear. Can I offer you a 50% discount to have you stay
with us?",

```

...

```

"BeginOffsetMillis": 12180,
"EndOffsetMillis": 16960,
"Sentiment": "NEGATIVE",
"ParticipantRole": "AGENT"

```

...

```

"Content": "That is a very generous offer. And I accept.",

```

...

```

"BeginOffsetMillis": 17140,
"EndOffsetMillis": 19860,
"Sentiment": "POSITIVE",
"ParticipantRole": "CUSTOMER"

```

- Valores quantitativos de sentimento para toda a chamada:

```

"Sentiment": {
  "OverallSentiment": {
    "AGENT": 2.5,
    "CUSTOMER": 2.1
  },

```

- Valores quantitativos de sentimento por participante e por um quarto da chamada:

```
"SentimentByPeriod": {
  "QUARTER": {
    "AGENT": [
      {
        "Score": 0.0,
        "BeginOffsetMillis": 0,
        "EndOffsetMillis": 9862
      },
      {
        "Score": -5.0,
        "BeginOffsetMillis": 9862,
        "EndOffsetMillis": 19725
      },
      {
        "Score": 5.0,
        "BeginOffsetMillis": 19725,
        "EndOffsetMillis": 29587
      },
      {
        "Score": 5.0,
        "BeginOffsetMillis": 29587,
        "EndOffsetMillis": 39450
      }
    ],
    "CUSTOMER": [
      {
        "Score": -2.5,
        "BeginOffsetMillis": 0,
        "EndOffsetMillis": 10615
      },
      {
        "Score": 5.0,
        "BeginOffsetMillis": 10615,
        "EndOffsetMillis": 21230
      },
      {
        "Score": 2.5,
        "BeginOffsetMillis": 21230,
        "EndOffsetMillis": 31845
      },
      {
        "Score": 5.0,
        "BeginOffsetMillis": 31845,
```



```
        "EndOffsetMillis": 42460
      }
    ]
  }
}
```

Edição de PII

Veja como é a edição de PII na saída de transcrição.

```
"Content": "[PII], my name is [PII], how can I help?",
"Redaction": [{
  "Confidence": "0.9998",
  "Type": "NAME",
  "Category": "PII"
}]
```

Consulte mais informações em [Redação de PII em seu trabalho em lotes](#).

Identificação de idioma

Veja como é a identificação de idioma na saída de transcrição caso o recurso esteja habilitado.

```
"LanguageIdentification": [{
  "Code": "en-US",
  "Score": "0.8299"
}, {
  "Code": "en-NZ",
  "Score": "0.0728"
}, {
  "Code": "zh-TW",
  "Score": "0.0695"
}, {
  "Code": "th-TH",
  "Score": "0.0156"
}, {
  "Code": "en-ZA",
  "Score": "0.0121"
}]
```

No exemplo de saída acima, a identificação de idioma preencherá os códigos de idioma com pontuações de confiança. O resultado com a pontuação mais alta será selecionado como o código do idioma para transcrição. Consulte mais detalhes em [Identifying the dominant languages in your media](#).

Saída de análise pós-chamada compilada

Para resumir, parte do conteúdo é substituído por elipses na saída de transcrição a seguir.

Esse exemplo inclui um recurso opcional: resumo generativo de chamadas.

```
{
  "JobStatus": "COMPLETED",
  "LanguageCode": "en-US",
  "Transcript": [
    {
      "LoudnessScores": [
        78.63,
        78.37,
        77.98,
        74.18
      ],
      "Content": "[PII], my name is [PII], how can I help?",
      ...

      "Content": "Well, I would like to cancel my recipe subscription.",
      "IssuesDetected": [
        {
          "CharacterOffsets": {
            "Begin": 7,
            "End": 51
          }
        }
      ],
      ...

      "Content": "That's very sad to hear. Can I offer you a 50% discount to have you stay with us?",
      "Items": [
        ...
      ],
    }
  ]
}
```

```

    "Id": "649afe93-1e59-4ae9-a3ba-a0a613868f5d",
    "BeginOffsetMillis": 12180,
    "EndOffsetMillis": 16960,
    "Sentiment": "NEGATIVE",
    "ParticipantRole": "AGENT"
  },
  {
    "LoudnessScores": [
      80.22,
      79.48,
      82.81
    ],
    "Content": "That is a very generous offer. And I accept.",
    "Items": [
      ...
    ],
    "Id": "f9266cba-34df-4ca8-9cea-4f62a52a7981",
    "BeginOffsetMillis": 17140,
    "EndOffsetMillis": 19860,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "CUSTOMER"
  },
  {
    ...

    "Content": "Wonderful. I made all changes to your account and now this
discount is applied, please check.",
    "OutcomesDetected": [
      {
        "CharacterOffsets": {
          "Begin": 12,
          "End": 78
        }
      }
    ],
    ...

    "Content": "I will send an email with all the details to you today, and I
will call you back next week to follow up. Have a wonderful evening.",
    "Items": [
      ...
    ],

```

```
    "Id": "78cd0923-cafd-44a5-a66e-09515796572f",
    "BeginOffsetMillis": 31800,
    "EndOffsetMillis": 39450,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "AGENT"
  },
  {
    "LoudnessScores": [
      78.54,
      68.76,
      67.76
    ],
    "Content": "Thank you very much, sir. Goodbye.",
    "Items": [
      ...
    ],
    "Id": "5c5e6be0-8349-4767-8447-986f995af7c3",
    "BeginOffsetMillis": 40040,
    "EndOffsetMillis": 42460,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "CUSTOMER"
  }
],
...

"Categories": {
  "MatchedDetails": {
    "positive-resolution": {
      "PointsOfInterest": [
        {
          "BeginOffsetMillis": 40040,
          "EndOffsetMillis": 42460
        }
      ]
    }
  },
  "MatchedCategories": [
    "positive-resolution"
  ]
},
...

```

```
"ConversationCharacteristics": {
  "NonTalkTime": {
    "Instances": [],
    "TotalTimeMillis": 0
  },
  "Interruptions": {
    "TotalCount": 2,
    "TotalTimeMillis": 10700,
    "InterruptionsByInterrupter": {
      "AGENT": [
        {
          "BeginOffsetMillis": 26040,
          "DurationMillis": 5510,
          "EndOffsetMillis": 31550
        }
      ],
      "CUSTOMER": [
        {
          "BeginOffsetMillis": 770,
          "DurationMillis": 5190,
          "EndOffsetMillis": 5960
        }
      ]
    }
  },
  "TotalConversationDurationMillis": 42460,
  "Sentiment": {
    "OverallSentiment": {
      "AGENT": 2.5,
      "CUSTOMER": 2.1
    },
    "SentimentByPeriod": {
      "QUARTER": {
        "AGENT": [
          {
            "Score": 0.0,
            "BeginOffsetMillis": 0,
            "EndOffsetMillis": 9862
          },
          {
            "Score": -5.0,
            "BeginOffsetMillis": 9862,
            "EndOffsetMillis": 19725
          }
        ]
      }
    }
  }
}
```

```
        {
            "Score": 5.0,
            "BeginOffsetMillis": 19725,
            "EndOffsetMillis": 29587
        },
        {
            "Score": 5.0,
            "BeginOffsetMillis": 29587,
            "EndOffsetMillis": 39450
        }
    ],
    "CUSTOMER": [
        {
            "Score": -2.5,
            "BeginOffsetMillis": 0,
            "EndOffsetMillis": 10615
        },
        {
            "Score": 5.0,
            "BeginOffsetMillis": 10615,
            "EndOffsetMillis": 21230
        },
        {
            "Score": 2.5,
            "BeginOffsetMillis": 21230,
            "EndOffsetMillis": 31845
        },
        {
            "Score": 5.0,
            "BeginOffsetMillis": 31845,
            "EndOffsetMillis": 42460
        }
    ]
}
},
"TalkSpeed": {
    "DetailsByParticipant": {
        "AGENT": {
            "AverageWordsPerMinute": 150
        },
        "CUSTOMER": {
            "AverageWordsPerMinute": 167
        }
    }
}
```

```

    }
  },
  "TalkTime": {
    "DetailsByParticipant": {
      "AGENT": {
        "TotalTimeMillis": 32750
      },
      "CUSTOMER": {
        "TotalTimeMillis": 18010
      }
    },
    "TotalTimeMillis": 50760
  },
  "ContactSummary": { // Optional feature - Generative call summarization
    "AutoGenerated": {
      "OverallSummary": {
        "Content": "The customer initially wanted to cancel but the agent
convinced them to stay by offering a 50% discount, which the customer accepted after
reconsidering cancelling given the significant savings. The agent ensured the discount
was applied and said they would follow up to ensure the customer remained happy with
the revised subscription."
      }
    }
  },
  "AnalyticsJobDetails": {
    "Skipped": []
  },
  ...
}

```

Habilitar o resumo generativo de chamada

Note

Desenvolvido pelo Amazon Bedrock: AWS implementa a detecção [automática de abusos](#). Como o resumo pós-contato desenvolvido com IA generativa é criado no Amazon Bedrock, os usuários podem aproveitar ao máximo os controles implementados no Amazon Bedrock para reforçar a segurança, a proteção e o uso responsável de inteligência artificial.

Para usar o resumo generativo de chamada com um trabalho de análise pós-chamada, veja os exemplos a seguir:

AWS Management Console

No painel Resumo, habilite o resumo generativo de chamada para receber um resumo na saída.

Configure job - *optional* [Info](#)

Content removal

Content removal conceals information in the resulting transcript from your source audio file. Amazon Transcribe changes items in the transcript and does not modify the source audio.

PII redaction [Info](#)

Label the type of PII and also mask the content with the PII entity type in the transcription output. For example, (123) 456-7890 will be masked as [PHONE].

Vocabulary filtering [Info](#)

Vocabulary filtering can remove, mask or tag specified words in the final transcript.

Customization

Custom vocabulary [Info](#)

A custom vocabulary improves the accuracy of recognizing words and phrases specific to your use case.

Summarization

Generative call summarization [Info](#)

Generative call summarization provides a summary of the transcript, including important components of the conversation.

Categories


Create categories to classify calls. For example, you can create a category for all cancellation requests. When you run an analytics job, Amazon Transcribe applies that category to all calls that request cancellation.

Call analytics categories (1) [Info](#)

< 1 > 

	Name	Type	Created	Modified
<input type="radio"/>	CatchNegativeSentiment	POST_CALL	February 17 2023, 10:43 (UTC-08:00)	February 17 2023, 10:43 (UTC-08:00)

Habilitar o resumo generativo de chamada

If the above categories aren't relevant to your use case, you can create a new category. [Create a new category.](#) 

AWS CLI

Este exemplo usa o comando [start-call-analytics-job](#) e o parâmetro Settings com os subparâmetros Summarization. Para ter mais informações, consulte [StartCallAnalyticsJob](#).

```
aws transcribe start-call-analytics-job \  
--region us-west-2 \  
--call-analytics-job-name my-first-call-analytics-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-location s3://DOC-EXAMPLE-BUCKET/my-output-files/ \  
--data-access-role-arn arn:aws:iam::111122223333:role/ExampleRole \  
--channel-definitions ChannelId=0,ParticipantRole=AGENT  
ChannelId=1,ParticipantRole=CUSTOMER  
--settings '{"Summarization":{"GenerateAbstractiveSummary":true}}'
```

Aqui está outro exemplo que usa o comando [start-call-analytics-job](#) e um corpo de solicitação que permite o resumo desse trabalho.

```
aws transcribe start-call-analytics-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-call-analytics-job.json
```

O arquivo `my-call-analytics-job.json` contém o corpo da solicitação a seguir.

```
{  
  "CallAnalyticsJobName": "my-first-call-analytics-job",  
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputLocation": "s3://DOC-EXAMPLE-BUCKET/my-output-files/",  
  "ChannelDefinitions": [  
    {  
      "ChannelId": 0,  
      "ParticipantRole": "AGENT"  
    },  
    {
```

```

    "ChannelId": 1,
    "ParticipantRole": "CUSTOMER"
  }
],
"Settings": {
  "Summarization":{
    "GenerateAbstractiveSummary": true
  }
}
}

```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para iniciar um Call Analytics com a sumarização habilitada usando o método [start_call_analytics_job](#). Para ter mais informações, consulte [StartCallAnalyticsJob](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

```

from __future__ import print_function
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-call-analytics-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
output_location = "s3://DOC-EXAMPLE-BUCKET/my-output-files/"
data_access_role = "arn:aws:iam::111122223333:role/ExampleRole"
transcribe.start_call_analytics_job(
    CallAnalyticsJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    DataAccessRoleArn = data_access_role,
    OutputLocation = output_location,
    ChannelDefinitions = [
        {
            'ChannelId': 0,
            'ParticipantRole': 'AGENT'
        },
    ],

```

```
{
  'ChannelId': 1,
  'ParticipantRole': 'CUSTOMER'
},
Settings = {
  "Summarization":
  {
    "GenerateAbstractiveSummary": true
  }
}
)

while True:
  status = transcribe.get_call_analytics_job(CallAnalyticsJobName = job_name)
  if status['CallAnalyticsJob']['CallAnalyticsJobStatus'] in ['COMPLETED', 'FAILED']:
    break
  print("Not ready yet...")
  time.sleep(5)
  print(status)
```

Análise de chamadas em tempo real

O Call Analytics em tempo real fornece informações em tempo real que podem ser usadas para resolver problemas e reduzir a tensão à medida que ela aumenta.

Os seguintes insights estão disponíveis com o Call Analytics em tempo real:

- [Eventos de categoria](#) que usam regras para sinalizar palavras-chave e frases específicas; os eventos de categoria podem ser usados para criar [alertas em tempo real](#).
- [Detecção de problemas](#), que identifica os problemas falados em cada segmento de áudio.
- [Identificação de PII \(dados confidenciais\)](#) na transcrição de texto.
- [Edição de PII \(dados confidenciais\)](#) da transcrição de texto.
- [Análise de sentimentos](#) para cada segmento da fala.

Além da análise de chamadas em tempo real, também Amazon Transcribe pode realizar [análises pós-chamada](#) em seu stream de mídia. Você pode incluir análises pós-chamada na solicitação de análise de chamadas em tempo real usando o parâmetro [PostCallAnalyticsSettings](#).

Insights em tempo real

Esta seção detalha os insights disponíveis para transcrições do Call Analytics em tempo real.

Eventos de categoria

Com os eventos de categoria, você pode fazer a correspondência da transcrição com uma palavra-chave ou frase exata. Por exemplo, se você definir um filtro para a frase “Quero falar com o gerente”, Amazon Transcribe filtre exatamente essa frase.

Veja a seguir um [exemplo de saída](#).

Para obter mais informações sobre a criação de categorias de análise de chamadas em tempo real, consulte [Criar categorias para transcrições em tempo real](#).

Tip

Os eventos de categoria permitem que você defina alertas em tempo real. Consulte [Criar alertas em tempo real para correspondências de categorias](#) para obter mais informações.

Detecção de problemas

A detecção de problemas fornece resumos sucintos dos problemas detectados em cada segmento de áudio. Com o recurso de detecção de problemas, é possível:

- Reduzir a necessidade de anotações manuais durante e após as chamadas.
- Melhorar a eficiência dos atendentes, permitindo que eles respondam mais rapidamente aos clientes.

Note

A detecção de problemas é compatível com esses dialetos do idioma inglês: australiano (en-AU), britânico (en-GB) e dos EUA (en-US).

O recurso de detecção de problemas funciona em todas as áreas e setores empresariais, e é baseado no contexto. Ele funciona out-of-the-box e, portanto, não oferece suporte à personalização, como treinamento de modelos ou categorias personalizadas.

Veja a seguir alguns exemplos do que você pode fazer com categorias personalizadas:

- Identificar problemas que merecem atenção imediata sinalizando e rastreando conjuntos específicos de palavras-chave.
- Monitorar a conformidade; por exemplo, um atendente que está falando (ou omitindo) uma frase específica.
- Sinalizar palavras e frases específicas em tempo real; você pode então definir sua correspondência de categoria para definir um alerta imediato. Por exemplo, se você criar uma categoria de análise de chamadas em tempo real para um cliente que diz “falar com um gerente”, será possível definir um [alerta de evento](#) para essa correspondência de categoria em tempo real que notifique o gerente de plantão.

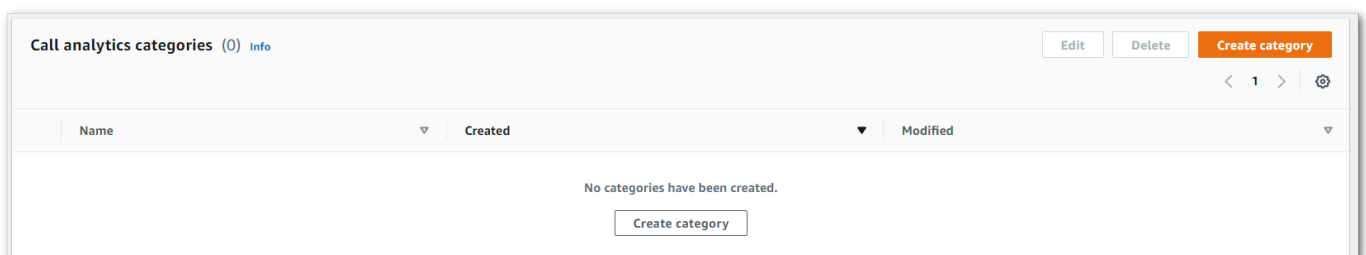
Categorias pós-chamada versus categorias em tempo real

Ao criar uma categoria, você pode especificar se deseja que ela seja criada como uma categoria pós-chamada (POST_CALL) ou como uma categoria em tempo real (REAL_TIME). Se não for especificada uma opção, a categoria será criada como uma categoria de pós-chamada por padrão. As correspondências de categorias em tempo real podem ser usadas para criar alertas em tempo real. Para ter mais informações, consulte [Criar alertas em tempo real para correspondências de categorias](#).

Para criar uma categoria de análise de chamadas em tempo real, você pode usar o AWS Management Console, a AWS CLI ou os SDKS da AWS ; veja seguintes exemplos:

AWS Management Console

1. No painel de navegação, em Amazon Transcribe, escolha Análise de Amazon Transcribe chamadas.
2. Escolha Categorias de análise de chamadas, que abrirá a página Categorias de análise de chamadas. Selecione o botão Criar categoria.



- Agora você está na página Criar categoria. Insira um nome para a categoria e escolha “Análise de chamadas em tempo real” no menu suspenso Tipo de categoria.

Category settings

Category name
MyCategory
The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, ., _ , and - (hyphen).

Category type [Info](#)
Choose category type ▲

Batch call analytics

Real time call analytics

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Template type [Info](#)
Choose the template for the category that most closely matches the one you want to create.

Choose a template ▼

- Você pode escolher um modelo para criar a categoria ou criar uma do zero.

Se estiver usando um modelo: selecione Usar um modelo (recomendado), escolha o modelo desejado e selecione Criar categoria.

Category settings

Category name
MyCategory
The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, ., _ , and - (hyphen).

Category type [Info](#)
Real time call analytics ▼

Category creation method [Info](#)

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Template type [Info](#)
Choose the template for the category that most closely matches the one you want to create.

Choose a template ▲

Customer content is negative and mentioned manager

- Se estiver criando uma categoria personalizada: selecione Criar do zero.

Create category [Info](#)

Category settings

Category name

The name can be up to 200 characters long. Valid characters are a-z, A-Z, 0-9, -, _ , and - (hyphen).

Category creation method [Info](#)

Use a template (recommended)
Use a template to edit predefined rules.

Create from scratch
If you know the rules that you want to define, choose this option.

Rules

All the rule conditions must be met for a transcription job to be classified in this category.

▼ Rule 1 Delete rule

Rule type [Info](#)
Choose the rule that you want to define.

Add rule

You can add up to 19 more rules.

- Adicione regras à categoria usando o menu suspenso. Você pode adicionar até vinte regras por categoria. Com as transcrições do Call Analytics em tempo real, você só pode incluir regras que envolvam correspondências de conteúdo da transcrição. Todas as correspondências são sinalizadas em tempo real.

Rules
All the rule conditions must be met for a transcription job to be classified in this category.

▼ **Rule 1** Delete rule

Rule type [Info](#)
Choose the rule that you want to define.

Choose a rule type ▲

Transcript content match
Trigger the rule when the speaker says the words or phrases that you specify.

Add rule

You can add up to 19 more rules.

7. Veja um exemplo de uma categoria com uma regra: um cliente que diz “falar com um gerente” a qualquer momento da chamada.

Rules
All the rule conditions must be met for a transcription job to be classified in this category.

▼ **Rule 1** Delete rule

When any of the words were mentioned during the entire call when the speaker was customer.

Rule type [Info](#)
Choose the rule that you want to define.

Transcript content match ▼

Logic [Info](#)
Define the conditions that must be met.

When any of the words were mentioned ▼

during the entire call ▼

when the speaker was customer ▼

Words or phrases [Info](#)
Enter the words or phrases that you want to look for in the transcript. You can enter up to 100 words or phrases.

Speak to a manager Add a new word or phrase

The word or phrase can be up to 2,000 characters.

Add rule

You can add up to 19 more rules.

8. Quando terminar de adicionar regras à sua categoria, escolha Criar categoria.

AWS CLI

Este exemplo usa o comando [create-call-analytics-category](#). Para obter mais informações, consulte [CreateCallAnalyticsCategory](#), [CategoryProperties](#) e [Rule](#).

O seguinte exemplo cria uma categoria com a regra:

- O cliente pronunciou a frase “falar com o gerente” em qualquer momento da chamada.

Este exemplo usa o comando [create-call-analytics-category](#) e um corpo de solicitação que adiciona uma regra à categoria.

```
aws transcribe create-call-analytics-category \  
--cli-input-json file://filepath/my-first-analytics-category.json
```

O arquivo `my-first-analytics-category.json` contém o corpo da solicitação a seguir.

```
{  
  "CategoryName": "my-new-real-time-category",  
  "InputType": "REAL_TIME",  
  "Rules": [  
    {  
      "TranscriptFilter": {  
        "Negate": false,  
        "Targets": [  
          "speak to the manager"  
        ],  
        "TranscriptFilterType": "EXACT"  
      }  
    }  
  ]  
}
```

AWS SDK for Python (Boto3)

Este exemplo usa o AWS SDK for Python (Boto3) para criar uma categoria usando os `Rules` argumentos `CategoryName` e para o método [create_call_analytics_category](#). Para obter mais informações, consulte [CreateCallAnalyticsCategory](#), [CategoryProperties](#) e [Rule](#).

Para ver exemplos adicionais de uso dos AWS SDKs, incluindo exemplos específicos de recursos, cenários e entre serviços, consulte o capítulo. [Exemplos de código para o Amazon Transcribe usando SDKs AWS](#)

O seguinte exemplo cria uma categoria com a regra:

- O cliente pronunciou a frase “falar com o gerente” em qualquer momento da chamada.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
category_name = "my-new-real-time-category"
transcribe.create_call_analytics_category(
    CategoryName = category_name,
    InputType = "REAL_TIME",
    Rules = [
        {
            'TranscriptFilter': {
                'Negate': False,
                'Targets': [
                    'speak to the manager'
                ],
                'TranscriptFilterType': 'EXACT'
            }
        }
    ]
)

result = transcribe.get_call_analytics_category(CategoryName = category_name)
print(result)
```

Critérios de regras para categorias de análise de chamadas em tempo real

Esta seção descreve os tipos de regras personalizadas REAL_TIME que podem ser criadas usando a operação de API [CreateCallAnalyticsCategory](#).

A detecção de problemas ocorre automaticamente, então não é necessário criar regras ou categorias para sinalizar problemas.

Observe que somente as correspondências de palavras-chave podem ser usadas com transcrições do Call Analytics em tempo real. Se você quiser criar categorias que incluam interrupções, silêncio ou sentimento, consulte [Critérios de regras para categorias de análise pós-chamada](#).

Correspondência de palavras-chave

As regras que usam palavras-chave (tipo de dados [TranscriptFilter](#)) foram criadas para corresponder a:

- Palavras ou frases personalizadas faladas pelo atendente, pelo cliente ou por ambos
- Palavras ou frases personalizadas não ditas pelo atendente, pelo cliente ou por ambos
- Palavras ou frases personalizadas que ocorrem em um período específico

Veja abaixo um exemplo dos parâmetros disponíveis com [TranscriptFilter](#):

```
"TranscriptFilter": {
  "AbsoluteTimeRange": {
    Specify the time frame, in milliseconds, when the match should occur
  },
  "RelativeTimeRange": {
    Specify the time frame, in percentage, when the match should occur
  },
  "Negate": Specify if you want to match the presence or absence of your custom keywords,
  "ParticipantRole": Specify if you want to match speech from the agent, the customer, or both,
  "Targets": [ The custom words and phrases you want to match ],
  "TranscriptFilterType": Use this parameter to specify an exact match for the specified targets
}
```

Consulte [CreateCallAnalyticsCategory](#) e [TranscriptFilter](#) para obter mais informações sobre esses parâmetros e os valores válidos associados a cada um.

Análise pós-chamada com transcrições em tempo real

A análise pós-chamada é um recurso opcional disponível com transcrições do Call Analytics em tempo real. Além dos [insights de análise em tempo real](#) padrão, a análise pós-chamada fornece o seguinte:

- Itens de ação: lista todos os itens de ação identificados na chamada.

- Interrupções: mede se e quando um participante interrompe o outro no meio da frase.
- Problemas: fornece os problemas identificados na chamada.
- Volume: mede o volume em que cada participante está falando.
- Tempo sem conversa: mede períodos que não contêm fala.
- Resultados: fornece o resultado, ou a resolução, identificado na chamada.
- Velocidade da conversa: mede a velocidade com que os dois participantes estão falando.
- Tempo de conversa: mede o tempo de fala (em milissegundos) de cada participante durante a chamada.

Quando ativada, a análise pós-chamada de um stream de áudio produz uma transcrição semelhante à [análise pós-chamada de um arquivo de áudio](#) e a armazena no Amazon S3 bucket especificado em `OutputLocation`. Além disso, a análise pós-chamada grava seu stream de áudio e o salva como um arquivo de áudio (WAVformato) no mesmo Amazon S3 bucket. Se você ativar a redação, uma transcrição editada e um arquivo de áudio editado também serão armazenados no bucket especificado. A habilitação da análise pós-chamada para o fluxo de áudio produz entre dois e quatro arquivos, conforme descrito aqui:

- Se a edição não estiver habilitada, os arquivos de saída serão:
 1. Uma transcrição sem oclusões
 2. Um arquivo de áudio sem oclusões
- Se a edição estiver habilitada sem a opção sem oclusões (`redacted`), os arquivos de saída serão:
 1. Uma transcrição editada
 2. Um arquivo de áudio editado
- Se a edição estiver habilitada com a opção sem oclusões (`redacted_and_unredacted`), os arquivos de saída serão:
 1. Uma transcrição editada
 2. Um arquivo de áudio editado
 3. Uma transcrição sem oclusões
 4. Um arquivo de áudio sem oclusões

Observe que, se você habilitar a análise pós-chamada ([PostCallAnalyticsSettings](#)) com sua solicitação e estiver usando FLAC ou a mídia OPUS-OGG, você não receberá `loudnessScore` na

transcrição e nenhuma gravação de áudio do fluxo será criada. O Transcribe também pode não ser capaz de fornecer análises pós-chamada para fluxos de áudio de longa duração com duração superior a 90 minutos.

Para ter mais informações sobre os insights disponíveis com a análise pós-chamada de fluxos de áudio, consulte a seção [Insights de análise pós-chamada](#).

Tip

Se você habilitar a análise pós-chamada com sua solicitação de análise de chamadas em tempo real, todas as categorias POST_CALL e REAL - TIME serão aplicadas à transcrição da análise pós-chamada.

Habilitar a análise pós-chamada

Para habilitar a análise pós-chamada, você deve incluir o parâmetro [PostCallAnalyticsSettings](#) na solicitação de análise de chamadas em tempo real. Os seguintes parâmetros devem ser incluídos quando PostCallAnalyticsSettings estiver habilitado:

- **OutputLocation**: o Amazon S3 repositório em que você deseja que sua transcrição pós-chamada seja armazenada.
- **DataAccessRoleArn**: o nome do recurso da Amazon (ARN) da função do Amazon S3 que tem permissões para acessar o bucket do Amazon S3 especificado. Observe que você também deve usar a [política de confiança para análises em tempo real](#).

Se desejar uma versão editada da transcrição, você pode incluir ContentRedactionOutput ou ContentRedactionType na solicitação. Para obter mais informações sobre esses parâmetros, consulte [StartCallAnalyticsStreamTranscription](#) na Referência da API.

Para iniciar uma transcrição do Call Analytics em tempo real com a análise pós-chamada ativada, você pode usar o AWS Management Console(somente demonstração), HTTP/2 ou. WebSockets Para ver exemplos, consulte [Iniciar uma transcrição do Call Analytics em tempo real](#).

⚠ Important

Atualmente, a AWS Management Console única oferece uma demonstração para análise de chamadas em tempo real com exemplos de áudio pré-carregados. Se quiser usar seu próprio áudio, você deve usar a API (HTTP/2 ou um SDK). WebSockets

Exemplo de saída de análise pós-chamada

As transcrições pós-chamada são exibidas em turn-by-turn formato por segmento. Elas incluem características da chamada, sentimento, resumo da chamada, detecção de problemas e (opcionalmente) edição de PII. Se alguma das categorias de pós-chamada corresponder ao conteúdo de áudio, elas também estarão presentes na saída.

Para aumentar a precisão e personalizar ainda mais as transcrições de acordo com seu caso de uso, como incluir termos específicos do setor, adicione [vocabulários personalizados](#) ou [modelos de idioma personalizados](#) à solicitação de análise de chamadas. Para mascarar, remover ou marcar palavras que você não deseja nos resultados da transcrição, como palavões, adicione a [filtragem de vocabulário](#).

Veja um exemplo compilado de saída de análise pós-chamada:

```
{
  "JobStatus": "COMPLETED",
  "LanguageCode": "en-US",
  "AccountId": "1234567890",
  "Channel": "VOICE",
  "Participants": [{
    "ParticipantRole": "AGENT"
  },
  {
    "ParticipantRole": "CUSTOMER"
  }
  ],
  "SessionId": "12a3b45c-de6f-78g9-0123-45h6ab78c901",
  "ContentMetadata": {
    "Output": "Raw"
  }
  "Transcript": [{
    "LoudnessScores": [
      78.63,
      78.37,
```

```
    77.98,
    74.18
  ],
  "Content": "[PII], my name is [PII], how can I help?",
  ...

  "Content": "Well, I would like to cancel my recipe subscription.",
  "IssuesDetected": [{
    "CharacterOffsets": {
      "Begin": 7,
      "End": 51
    }
  }],
  ...

  "Content": "That's very sad to hear. Can I offer you a 50% discount to have you
stay with us?",
  "Id": "649afe93-1e59-4ae9-a3ba-a0a613868f5d",
  "BeginOffsetMillis": 12180,
  "EndOffsetMillis": 16960,
  "Sentiment": "NEGATIVE",
  "ParticipantRole": "AGENT"
},
{
  "LoudnessScores": [
    80.22,
    79.48,
    82.81
  ],
  "Content": "That is a very generous offer. And I accept.",
  "Id": "f9266cba-34df-4ca8-9cea-4f62a52a7981",
  "BeginOffsetMillis": 17140,
  "EndOffsetMillis": 19860,
  "Sentiment": "POSITIVE",
  "ParticipantRole": "CUSTOMER"
},
  ...

  "Content": "Wonderful. I made all changes to your account and now this discount
is applied, please check.",
  "OutcomesDetected": [{
    "CharacterOffsets": {
```

```

        "Begin": 12,
        "End": 78
    }
}],

    ...

    "Content": "I will send an email with all the details to you today, and I will
call you back next week to follow up. Have a wonderful evening.",
    "Id": "78cd0923-cafd-44a5-a66e-09515796572f",
    "BeginOffsetMillis": 31800,
    "EndOffsetMillis": 39450,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "AGENT"
},
{
    "LoudnessScores": [
        78.54,
        68.76,
        67.76
    ],
    "Content": "Thank you very much, sir. Goodbye.",
    "Id": "5c5e6be0-8349-4767-8447-986f995af7c3",
    "BeginOffsetMillis": 40040,
    "EndOffsetMillis": 42460,
    "Sentiment": "POSITIVE",
    "ParticipantRole": "CUSTOMER"
}
],

    ...

"Categories": {
    "MatchedDetails": {
        "positive-resolution": {
            "PointsOfInterest": [{
                "BeginOffsetMillis": 40040,
                "EndOffsetMillis": 42460
            }]
        }
    },
    "MatchedCategories": [
        "positive-resolution"
    ]
}

```

```
},
...
"ConversationCharacteristics": {
  "NonTalkTime": {
    "Instances": [],
    "TotalTimeMillis": 0
  },
  "Interruptions": {
    "TotalCount": 2,
    "TotalTimeMillis": 10700,
    "InterruptionsByInterrupter": {
      "AGENT": [{
        "BeginOffsetMillis": 26040,
        "DurationMillis": 5510,
        "EndOffsetMillis": 31550
      }],
      "CUSTOMER": [{
        "BeginOffsetMillis": 770,
        "DurationMillis": 5190,
        "EndOffsetMillis": 5960
      }]
    }
  },
  "TotalConversationDurationMillis": 42460,
  "Sentiment": {
    "OverallSentiment": {
      "AGENT": 2.5,
      "CUSTOMER": 2.1
    },
    "SentimentByPeriod": {
      "QUARTER": {
        "AGENT": [{
          "Score": 0.0,
          "BeginOffsetMillis": 0,
          "EndOffsetMillis": 9862
        }],
        {
          "Score": -5.0,
          "BeginOffsetMillis": 9862,
          "EndOffsetMillis": 19725
        }
      }
    }
  }
}
```

```
        "Score": 5.0,
        "BeginOffsetMillis": 19725,
        "EndOffsetMillis": 29587
    },
    {
        "Score": 5.0,
        "BeginOffsetMillis": 29587,
        "EndOffsetMillis": 39450
    }
],
"CUSTOMER": [{
    "Score": -2.5,
    "BeginOffsetMillis": 0,
    "EndOffsetMillis": 10615
},
{
    "Score": 5.0,
    "BeginOffsetMillis": 10615,
    "EndOffsetMillis": 21230
},
{
    "Score": 2.5,
    "BeginOffsetMillis": 21230,
    "EndOffsetMillis": 31845
},
{
    "Score": 5.0,
    "BeginOffsetMillis": 31845,
    "EndOffsetMillis": 42460
}
]
}
},
"TalkSpeed": {
    "DetailsByParticipant": {
        "AGENT": {
            "AverageWordsPerMinute": 150
        },
        "CUSTOMER": {
            "AverageWordsPerMinute": 167
        }
    }
},
},
```

```
"TalkTime": {
  "DetailsByParticipant": {
    "AGENT": {
      "TotalTimeMillis": 32750
    },
    "CUSTOMER": {
      "TotalTimeMillis": 18010
    }
  },
  "TotalTimeMillis": 50760
},
...
}
```

Iniciar uma transcrição do Call Analytics em tempo real

Antes de iniciar uma transcrição do Call Analytics em tempo real, você deve criar todas as [categorias](#) que Amazon Transcribe deseja combinar em sua chamada.

Note


As transcrições do Call Analytics não podem ser combinadas retroativamente com novas categorias. Somente as categorias que você cria antes de iniciar uma transcrição do Call Analytics podem ser aplicadas a essa saída de transcrição.

Se você tiver criado uma ou mais categorias e o áudio corresponder a todas as regras de pelo menos uma das categorias, o Amazon Transcribe marcará a saída com as categorias correspondentes. Se você optar por não usar categorias ou se o áudio não corresponder às regras especificadas nas categorias, sua transcrição não será sinalizada.

Para incluir a análise pós-chamada na transcrição do Call Analytics em tempo real, você deve fornecer um bucket do Amazon S3 na solicitação usando o parâmetro `OutputLocation`. Você também deve incluir um `DataAccessRoleArn` que tenha permissões de gravação no bucket especificado. Uma transcrição separada é produzida e armazenada no bucket especificado após a conclusão da sessão de streaming do Call Analytics em tempo real.

Com o Call Analytics em tempo real, você também tem a opção de criar alertas de categoria em tempo real. Consulte [Criar alertas em tempo real para correspondências de categorias](#) para obter instruções.


Para iniciar uma transcrição do Call Analytics em tempo real, você pode usar o AWS Management Console HTTP/2 ou WebSockets; veja exemplos a seguir:

 Important

Atualmente, a AWS Management Console única oferece uma demonstração para análise de chamadas em tempo real com exemplos de áudio pré-carregados. Se quiser usar seu próprio áudio, você deve usar a API (HTTP/2 ou um SDK). WebSockets

AWS Management Console

Use o procedimento a seguir para iniciar uma solicitação do Call Analytics. As chamadas que correspondem a todas as características definidas por uma categoria são rotuladas com essa categoria.

 Note

Somente uma demonstração está disponível no AWS Management Console. Para iniciar uma transcrição personalizada de análise em tempo real, você deve usar a [API](#).

1. No painel de navegação, em Análise de Amazon Transcribe chamadas, escolha Analisar uma chamada em tempo real.

Amazon Transcribe > Real-time Analytics

Real-time Analytics [info](#)

Transcribe Real-time Call Analytics combines powerful speech-to-text and natural language processing (NLP) models that are trained specifically to understand customer service and sales calls. With Transcribe Call Analytics, developers can get a redacted and unredacted transcript, and insights such as customer and agent sentiment, detected issues, and supervisor alerts during the live call.

How it works
This demo experience has been configured to use preloaded audio examples of customer-agent interactions. Before starting the demo, you can optionally create categories in the Category Management page and update content redaction settings under the advance settings

Step 1: Specify input audio

Input audio file

Insurance complaints (en-US) ▼

▶ 00:00/00:00

Step 2: Review call categories - optional

Categorize your calls based on custom keywords or phrases.

[View categories](#)

Step 3: Configure output - optional

Apply content redaction settings to your calls.

[Configure advanced settings](#)

Post-call Analytics
Post-call analytics enabled with real-time analytics provides consolidated transcript and audio backup, with the associated analytics, along with further insights such as call summaries and conversation characteristics like non-talk time, interruptions, loudness, and talk speed, after the end of the call in the provided Amazon S3 bucket.

Post-call Analytics

[Start streaming](#)

- Para Etapa 1: especificar áudio de entrada, escolha um arquivo de teste de demonstração no menu suspenso.



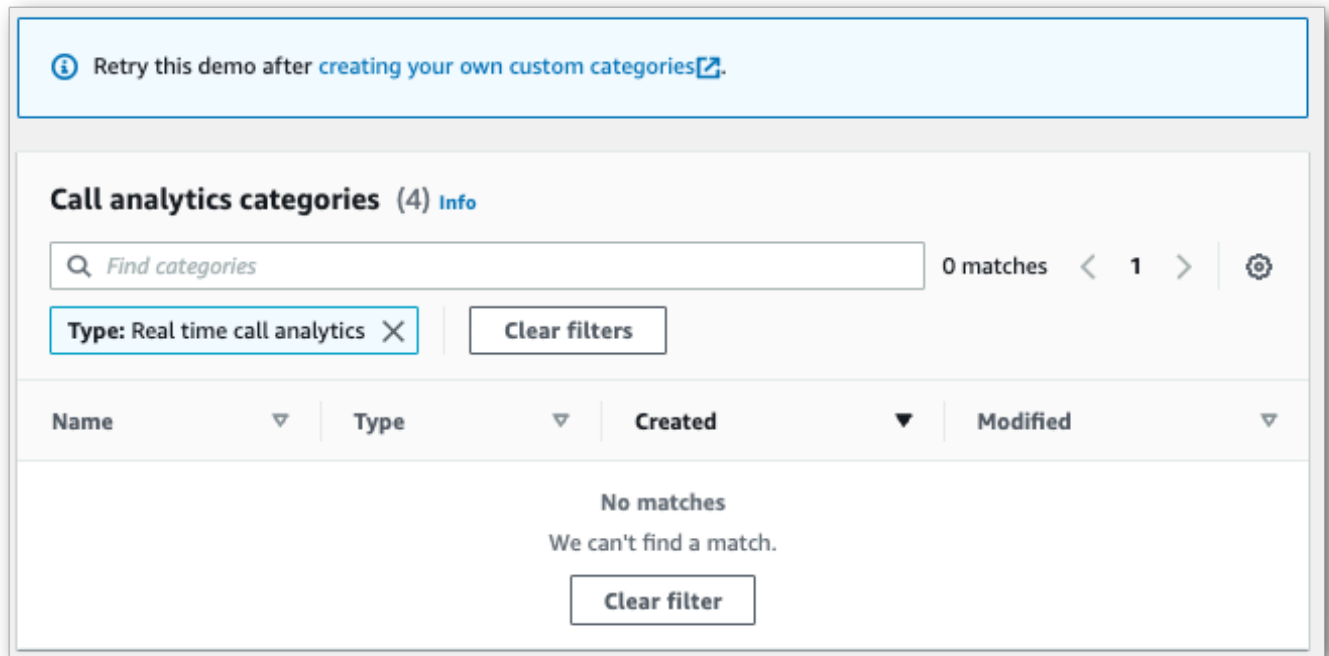
Step 1: Specify input audio

Input audio file

Insurance complaints (en-US)	▲
Insurance complaints (en-US)	✓
Hospitality complaints (en-US)	

- Para a Etapa 2: revisar categorias de chamada, você tem a opção de revisar as categorias de análise de chamadas em tempo real que criou anteriormente. Todas as categorias de análise de chamadas em tempo real são aplicadas à transcrição.

Escolher Exibir categorias abre um novo painel que mostra as categorias existentes de análise de chamadas em tempo real e fornece um link para criar outras.



4. Para Etapa 3: configurar entrada e saída, você tem a opção de aplicar configurações adicionais.

Escolher Definir configurações avançadas abre um novo painel onde você pode especificar as configurações de edição de conteúdo.

Use the following options to identify or redact content from your transcript. Other settings such as Custom Vocabulary, Custom Language Models, Partial results stabilization, Vocabulary Filtering are available through the API, SDK, CLI

▼ Content removal

PII Identification & redaction [Info](#)
 Identify or redact one or more types of personally identifiable information (PII) in your transcript

Select PII detection type

Identification only
 Label the type of PII identified but not redact it in the transcription output

Identification & redaction
 Label the type of PII and also mask the content with the PII entity type in the transcription output.
 For example, (123)456-7890 will be masked as [PHONE]

Select PII entity types (11 of 11 selected)

Select All

Financial (6 of 6 selected)

<input checked="" type="checkbox"/> BANK_ACCOUNT_NUMBER	<input checked="" type="checkbox"/> BANK_ROUTING	<input checked="" type="checkbox"/> CREDIT_DEBIT_NUMBER
<input checked="" type="checkbox"/> CREDIT_DEBIT_CVV	<input checked="" type="checkbox"/> CREDIT_DEBIT_EXPIRY	<input checked="" type="checkbox"/> PIN

Personal (5 of 5 selected)

<input checked="" type="checkbox"/> NAME	<input checked="" type="checkbox"/> ADDRESS	<input checked="" type="checkbox"/> PHONE
<input checked="" type="checkbox"/> EMAIL	<input checked="" type="checkbox"/> SSN	

i The updates that you make here will only be applied when you start stream again.

Cancel
Save

Depois de fazer todas as seleções, escolha Salvar para retornar à página principal.

5. Para aplicar análises adicionais, você pode ativar a Análise pós-chamada. Isso fornece a mesma análise de uma transcrição de análise pós-chamada, incluindo interrupções, volume, tempo sem conversa, velocidade da conversa, tempo de conversa, problemas, itens de ação e resultados. A saída da análise pós-chamada é armazenada em um arquivo separado da transcrição do Call Analytics em tempo real.

Post-call Analytics
 Post-call analytics enabled with real-time analytics provides consolidated transcript and audio backup, with the associated analytics, along with further insights such as call summaries and conversation characteristics like non-talk time, interruptions, loudness, and talk speed, after the end of the call in the provided Amazon S3 bucket.

Post-call Analytics

Se você aplicar a análise pós-chamada, deverá especificar um destino Amazon S3 de arquivo de saída e uma IAM função. Se preferir, você poderá criptografar a saída.

Post-call Analytics
Post-call analytics enabled with real-time analytics provides consolidated transcript and audio backup, with the associated analytics, along with further insights such as call summaries and conversation characteristics like non-talk time, interruptions, loudness, and talk speed, after the end of the call in the provided Amazon S3 bucket.

Post-call Analytics

Output file destination on S3 [Info](#)
Choose the location to store the output of the post-call analytics. If you input a location in an Amazon S3 bucket that doesn't yet exist, it will be created for you.

Resource URI

Format: s3://bucket, s3://bucket/prefix/, or s3://bucket/prefix/object.

Encryption [Info](#)

IAM role [Info](#)

[Create an IAM role](#) that grants access to the output bucket and KMS key (if specified) with the trust policy shown below
▶ Trust Policy

6. Selecione Start streaming (Iniciar transmissão).

Fluxo HTTP/2

Este exemplo cria uma solicitação HTTP/2 com o Call Analytics habilitado. Para obter mais informações sobre como usar o streaming HTTP/2 com Amazon Transcribe, consulte [Configurar um fluxo de HTTP/2](#). Para obter mais detalhes sobre parâmetros e cabeçalhos específicos de Amazon Transcribe, consulte [StartCallAnalyticsStreamTranscription](#).

Esse exemplo inclui [análises pós-chamada](#). Se não quiser análises pós-chamada, remova a seção `PostCallAnalyticsSettings` da solicitação.

Observe que o evento de configuração mostrado no exemplo a seguir precisa ser passado como o primeiro evento no fluxo.

```
POST /stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
X-Amz-Target: com.amazonaws.transcribe.Transcribe.StartCallAnalyticsStreamTranscription
Content-Type: application/vnd.amazon.eventstream
X-Amz-Content-Sha256: string
X-Amz-Date: 20220208T235959Z
Authorization: AWS4-HMAC-SHA256 Credential=access-key/20220208/us-west-2/transcribe/
aws4_request, SignedHeaders=content-type;host;x-amz-content-sha256;x-amz-date;x-amz-
target;x-amz-security-token, Signature=string
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
```

```
transfer-encoding: chunked

{
  "AudioStream": {
    "AudioEvent": {
      "AudioChunk": blob
    },
    "ConfigurationEvent": {
      "ChannelDefinitions": [
        {
          "ChannelId": 0,
          "ParticipantRole": "AGENT"
        },
        {
          "ChannelId": 1,
          "ParticipantRole": "CUSTOMER"
        }
      ],
      "PostCallAnalyticsSettings": {
        "OutputLocation": "s3://DOC-EXAMPLE-BUCKET/my-output-files/",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole"
      }
    }
  }
}
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

WebSocket stream

Este exemplo cria uma URL pré-assinada que usa o Call Analytics em um WebSocket stream. Quebras de linha foram adicionadas para uma melhor leitura. Para obter mais informações sobre o uso de WebSocket fluxos com Amazon Transcribe, consulte [Configurando um WebSocket stream](#). Para obter mais detalhes sobre os parâmetros, consulte [StartCallAnalyticsStreamTranscription](#).

Esse exemplo inclui [análises pós-chamada](#). Se não quiser análises pós-chamada, remova a seção `PostCallAnalyticsSettings` da solicitação.

Observe que o evento de configuração mostrado no exemplo a seguir precisa ser passado como o primeiro evento no fluxo.

```

GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/call-analytics-stream-
transcription-websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=string
&X-Amz-SignedHeaders=content-type%3Bhost%3Bx-amz-date
&language-code=en-US
&media-encoding=flac
&sample-rate=16000

{
  "AudioStream": {
    "AudioEvent": {
      "AudioChunk": blob
    },
    "ConfigurationEvent": {
      "ChannelDefinitions": [
        {
          "ChannelId": 0,
          "ParticipantRole": "AGENT"
        },
        {
          "ChannelId": 1,
          "ParticipantRole": "CUSTOMER"
        }
      ],
      "PostCallAnalyticsSettings": {
        "OutputLocation": "s3://DOC-EXAMPLE-BUCKET/my-output-files/",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole"
      }
    }
  }
}

```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

i Tip

O HTTP/2 e os WebSocket exemplos acima incluem análises pós-chamada. Se não quiser análises pós-chamada, remova a seção `PostCallAnalyticsSettings` da solicitação. Se você habilitar `PostCallAnalyticsSettings`, deverá enviar um evento de configuração como o primeiro evento. O evento de configuração inclui configurações para `ChannelDenifitions` e `PostStreamAnalyticsSettings`, conforme mostrado nos exemplos anteriores.

Os dados binários são transmitidos como uma mensagem binária com `content-type application/octet-stream` e o evento de configuração é transmitido como uma mensagem de texto com `content-type application/json`.

Para ter mais informações, consulte [Configurar uma transcrição de streaming](#).

Criar alertas em tempo real para correspondências de categorias

Para configurar alertas em tempo real, você deve primeiro criar uma categoria

[TranscriptFilterType](#) com o sinalizador `REAL_TIME`. Esse sinalizador permite que a categoria seja aplicada às transcrições do Call Analytics em tempo real.

Para obter instruções sobre como criar uma categoria, consulte [Criar categorias para transcrições em tempo real](#).

Quando você inicia a transcrição do Call Analytics em tempo real, todas as categorias que têm o sinalizador `REAL_TIME` são aplicadas automaticamente ao resultado da transcrição em um nível de segmento. Se ocorrer uma correspondência `TranscriptFilterType`, ela aparecerá na seção `CategoryEvent` da transcrição. Em seguida, você pode usar esse parâmetro e os respectivos subparâmetros, `MatchedCategories` e `MatchedDetails`, para configurar alertas personalizados em tempo real.

Veja um exemplo de saída de transcrição do Call Analytics em tempo real para uma correspondência `CategoryEvent`:

```
"CategoryEvent": {
  "MatchedCategories": [ "shipping-complaint" ],
  "MatchedDetails": {
    "my package never arrived" : {
      "TimestampRanges": [
```

```
    {
      "BeginOffsetMillis": 19010,
      "EndOffsetMillis": 22690
    }
  ]
}
},
```

O exemplo anterior representa uma correspondência exata de texto com a fala “meu pacote não chegou”, que representa uma regra dentro da categoria “reclamação de envio”.

Você pode configurar o alerta em tempo real para incluir qualquer combinação dos parâmetros listados. Por exemplo, é possível configurar o alerta para incluir somente a frase para a qual houve correspondência (`MatchedDetails`) ou somente o nome da categoria (`MatchedCategories`). Ou você pode configurar o alerta para incluir todos os parâmetros.

A forma como você configura os alertas em tempo real depende das interfaces da organização e do tipo de alerta desejado. Por exemplo, você pode definir uma correspondência `CategoryEvent` para enviar uma notificação pop-up, um e-mail, uma mensagem de texto ou qualquer outro alerta que o sistema possa aceitar.

Saída de análise de chamadas em tempo real

As transcrições do Call Analytics em tempo real são exibidas em turn-by-turn formato por segmento. Elas incluem eventos de categoria, detecção de problemas, sentimento e identificação e edição de PII. Os eventos de categoria permitem que você defina alertas em tempo real. Consulte [Criar alertas em tempo real para correspondências de categorias](#) para obter mais informações.

Para aumentar a precisão e personalizar ainda mais as transcrições de acordo com seu caso de uso, como incluir termos específicos do setor, adicione [vocabulários personalizados](#) ou [modelos de idioma personalizados](#) à solicitação de análise de chamadas. Para mascarar, remover ou marcar palavras que você não deseja nos resultados da transcrição, como palavrões, adicione a [filragem de vocabulário](#).

As seções a seguir mostram exemplos de saída JSON para transcrições de análise de chamadas em tempo real.

Eventos de categoria

Veja como é uma correspondência de categoria na saída de transcrição. Este exemplo mostra que o áudio do carimbo de data e hora de 19.010 milissegundos para o carimbo de data e hora de 22.690 milissegundos corresponde à categoria “reclamação de rede”. Nesse caso, a categoria personalizada de “reclamação de rede” exigia que o cliente dissesse “problemas de rede” (correspondência exata de palavras).

```
"CategoryEvent": {
  "MatchedCategories": [
    "network-complaint"
  ],
  "MatchedDetails": {
    "network issues" : {
      "TimestampRanges": [
        {
          "BeginOffsetMillis": 9299375,
          "EndOffsetMillis": 7899375
        }
      ]
    }
  }
},
```

Detecção de problemas

Veja como é uma correspondência de detecção de problemas na saída de transcrição. Este exemplo mostra que o texto do caractere 26 ao caractere 62 descreve um problema.

```
"UtteranceEvent": {
  ...
  "Transcript": "Wang Xiulan I'm tired of the network issues my phone is having.",
  ...
  "IssuesDetected": [
    {
      "CharacterOffsets": {
        "BeginOffsetChar": 26,
        "EndOffsetChar": 62
      }
    }
  ]
}
```



```
},
```

Sentimento

Veja como é a análise de sentimentos na saída de transcrição.

```
"UtteranceEvent": {
  ...
  "Sentiment": "NEGATIVE",
  "Items": [{
    ...
```

Identificação de PII

Veja como é a identificação de PII na saída de transcrição.

```
"Entities": [
  {
    "Content": "Wang Xiulan",
    "Category": "PII",
    "Type": "NAME",
    "BeginOffsetMillis": 7999375,
    "EndOffsetMillis": 199375,
    "Confidence": 0.9989
  }
],
```

Edição de PII

Esta é a aparência da edição de PII na saída de transcrição.

```
"Content": "[NAME]. Hi, [NAME]. I'm [NAME] Happy to be helping you today.",
"Redaction": {
  "RedactedTimestamps": [
    {
      "BeginOffsetMillis": 32670,
      "EndOffsetMillis": 33343
    },
    {
      "BeginOffsetMillis": 33518,
      "EndOffsetMillis": 33858
    },
  ],
```

```

    {
      "BeginOffsetMillis": 34068,
      "EndOffsetMillis": 34488
    }
  ],
},

```

Saída compilada de análise de chamadas em tempo real

Por motivo de brevidade, alguns conteúdos são substituídos por elipses na saída de transcrição a seguir.

```

{
  "CallAnalyticsTranscriptResultStream": {
    "BadRequestException": {},
    "ConflictException": {},
    "InternalFailureException": {},
    "LimitExceededException": {},
    "ServiceUnavailableException": {},
    "UtteranceEvent": {
      "UtteranceId": "58c27f92-7277-11ec-90d6-0242ac120003",
      "ParticipantRole": "CUSTOMER",
      "IsPartial": false,
      "Transcript": "Wang Xiulan I'm tired of the network issues my phone is
having.",
      "BeginOffsetMillis": 19010,
      "EndOffsetMillis": 22690,
      "Sentiment": "NEGATIVE",
      "Items": [{
        "Content": "Wang",
        "BeginOffsetMillis": 379937,
        "EndOffsetMillis": 299375,
        "Type": "pronunciation",
        "Confidence": 0.9961,
        "VocabularyFilterMatch": false
      },
      {
        "Content": "Xiulan",
        "EndOffsetMillis": 5899375,
        "BeginOffsetMillis": 3899375,
        "Type": "pronunciation",
        "Confidence": 0.9961,
        "VocabularyFilterMatch": false
      }
    ]
  }
}

```

```
    },
    ...
    {
      "Content": "network",
      "EndOffsetMillis": 199375,
      "BeginOffsetMillis": 9299375,
      "Type": "pronunciation",
      "Confidence": 0.9961,
      "VocabularyFilterMatch": false
    },
    {
      "Content": "issues",
      "EndOffsetMillis": 7899375,
      "BeginOffsetMillis": 5999375,
      "Type": "pronunciation",
      "Confidence": 0.9961,
      "VocabularyFilterMatch": false
    },
    {
      "Content": "my",
      "EndOffsetMillis": 9199375,
      "BeginOffsetMillis": 7999375,
      "Type": "pronunciation",
      "Confidence": 0.9961,
      "VocabularyFilterMatch": false
    },
    {
      "Content": "phone",
      "EndOffsetMillis": 199375,
      "BeginOffsetMillis": 9299375,
      "Type": "pronunciation",
      "Confidence": 0.9961,
      "VocabularyFilterMatch": false
    },
    ...
  ],
  "Entities": [{
    "Content": "Wang Xiulan",
    "Category": "PII",
    "Type": "NAME",
    "BeginOffsetMillis": 7999375,
    "EndOffsetMillis": 199375,
    "Confidence": 0.9989
  }],
}],
```

```
    "IssuesDetected": [{
      "CharacterOffsets": {
        "BeginOffsetChar": 26,
        "EndOffsetChar": 62
      }
    }]
  },
  "CategoryEvent": {
    "MatchedCategories": [
      "network-complaint"
    ],
    "MatchedDetails": {
      "network issues" : {
        "TimestampRanges": [
          {
            "BeginOffsetMillis": 9299375,
            "EndOffsetMillis": 7899375
          }
        ]
      }
    }
  }
}
```

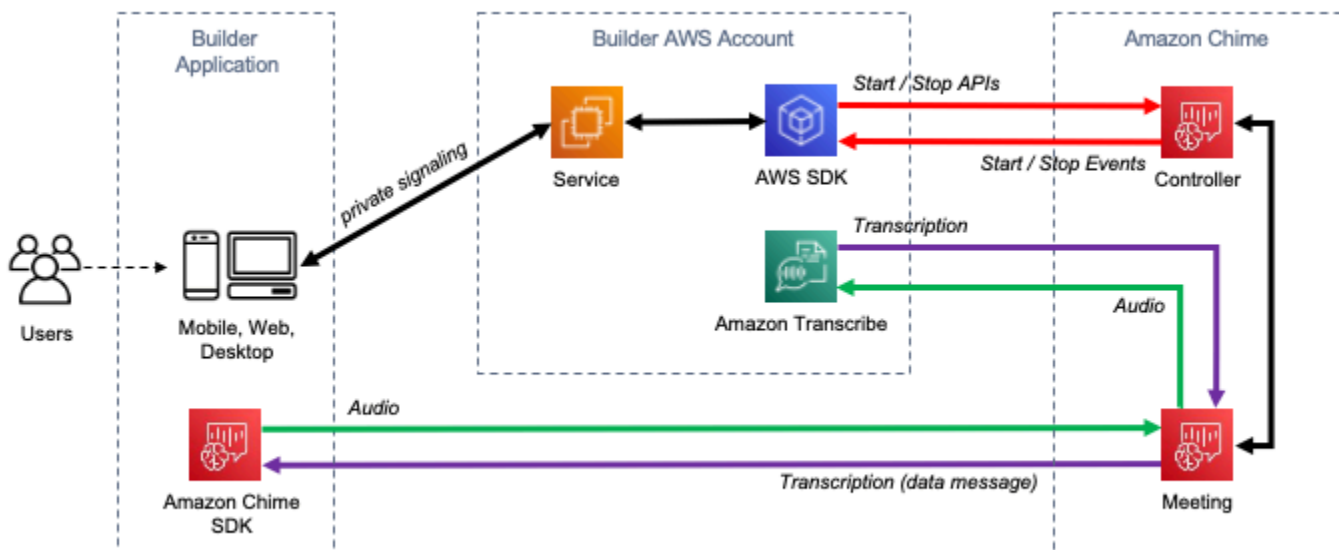
Transcrevendo suas Amazon Chime chamadas em tempo real

Amazon Transcribe é integrado ao Amazon Chime SDK, facilitando as transcrições em tempo real de suas Amazon Chime chamadas.


Quando você solicita uma transcrição usando a API do Amazon Chime SDK, Amazon Chime começa a transmitir áudio Amazon Transcribe e continua fazendo isso durante a chamada.

O Amazon Chime SDK usa seu algoritmo de “locutor ativo” para selecionar os dois principais locutores ativos e, em seguida, envia seu áudio para Amazon Transcribe dois canais separados por meio de um único stream. Os participantes da reunião recebem transcrições atribuídas pelo usuário por meio de mensagens de dados do Amazon Chime SDK. Você pode ver exemplos de entrega no [Guia do desenvolvedor do Amazon Chime SDK](#).

O fluxo de dados de uma Amazon Chime transcrição é representado no diagrama a seguir:



Para obter informações adicionais e instruções detalhadas sobre como configurar transcrições em tempo real, consulte Como [usar a transcrição ao vivo do Amazon Chime SDK](#) no Guia do desenvolvedor do Amazon Chime SDK. Para operações de API, consulte a [Referência da API do Amazon Chime SDK](#).

 Mergulhe mais fundo com o blog [AWS Machine Learning](#)

Para saber mais sobre como melhorar a precisão com transcrições em tempo real, consulte:

- [Amazon ChimeAs reuniões do SDK agora oferecem suporte à transcrição ao vivo com Amazon Transcribe e Amazon Transcribe Medical](#)
- [Amazon ChimeSDK para solução de telemedicina](#)

Exemplos de código para o Amazon Transcribe usando SDKs AWS

Os exemplos de código a seguir mostram como usar o Amazon Transcribe com AWS um kit de desenvolvimento de software (SDK).

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar funções de serviço específicas, é possível ver as ações contextualizadas em seus devidos cenários e exemplos entre serviços.

Cenários são exemplos de código que mostram como realizar uma tarefa específica chamando várias funções dentro do mesmo serviço.

Exemplos entre serviços são amostras de aplicações que funcionam em vários Serviços da AWS.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Exemplos de código

- [Ações para o Amazon Transcribe usando SDKs AWS](#)
 - [Use CreateVocabulary com um AWS SDK ou CLI](#)
 - [Use DeleteMedicalTranscriptionJob com um AWS SDK ou CLI](#)
 - [Use DeleteTranscriptionJob com um AWS SDK ou CLI](#)
 - [Use DeleteVocabulary com um AWS SDK ou CLI](#)
 - [Use GetTranscriptionJob com um AWS SDK ou CLI](#)
 - [Use GetVocabulary com um AWS SDK ou CLI](#)
 - [Use ListMedicalTranscriptionJobs com um AWS SDK ou CLI](#)
 - [Use ListTranscriptionJobs com um AWS SDK ou CLI](#)
 - [Use ListVocabularies com um AWS SDK ou CLI](#)
 - [Use StartMedicalTranscriptionJob com um AWS SDK ou CLI](#)
 - [Use StartStreamTranscriptionAsync com um AWS SDK ou CLI](#)
 - [Use StartTranscriptionJob com um AWS SDK ou CLI](#)
 - [Use UpdateVocabulary com um AWS SDK ou CLI](#)

- [Cenários para o Amazon Transcribe usando SDKs AWS](#)
 - [Crie e refine um vocabulário personalizado do Amazon Transcribe usando um SDK AWS](#)
 - [Transcreva áudio e obtenha dados de trabalho com o Amazon Transcribe usando um SDK AWS](#)
- [Exemplos de serviços cruzados para o Amazon Transcribe usando SDKs AWS](#)
 - [Criar uma aplicação Amazon Transcribe](#)
 - [Criar uma aplicação de transmissão do Amazon Transcribe](#)
 - [Converta texto em fala e vice-versa em texto usando um AWS SDK](#)

Ações para o Amazon Transcribe usando SDKs AWS

Os exemplos de código a seguir demonstram como realizar ações individuais do Amazon Transcribe com SDKs. AWS Esses trechos chamam a API do Amazon Transcribe e são trechos de código de programas maiores que devem ser executados no contexto. Cada exemplo inclui um link para GitHub, onde você pode encontrar instruções para configurar e executar o código.

Os exemplos a seguir incluem apenas as ações mais utilizadas. Para obter uma lista completa, consulte a [Referência da API do Amazon Transcribe](#).

Exemplos

- [Use CreateVocabulary com um AWS SDK ou CLI](#)
- [Use DeleteMedicalTranscriptionJob com um AWS SDK ou CLI](#)
- [Use DeleteTranscriptionJob com um AWS SDK ou CLI](#)
- [Use DeleteVocabulary com um AWS SDK ou CLI](#)
- [Use GetTranscriptionJob com um AWS SDK ou CLI](#)
- [Use GetVocabulary com um AWS SDK ou CLI](#)
- [Use ListMedicalTranscriptionJobs com um AWS SDK ou CLI](#)
- [Use ListTranscriptionJobs com um AWS SDK ou CLI](#)
- [Use ListVocabularies com um AWS SDK ou CLI](#)
- [Use StartMedicalTranscriptionJob com um AWS SDK ou CLI](#)
- [Use StartStreamTranscriptionAsync com um AWS SDK ou CLI](#)
- [Use StartTranscriptionJob com um AWS SDK ou CLI](#)
- [Use UpdateVocabulary com um AWS SDK ou CLI](#)

Use **CreateVocabulary** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `CreateVocabulary`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação no contexto no seguinte exemplo de código:

- [Criar e refinar um vocabulário personalizado](#)

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// Create a custom vocabulary using a list of phrases. Custom vocabularies
/// improve transcription accuracy for one or more specific words.
/// </summary>
/// <param name="languageCode">The language code of the vocabulary.</param>
/// <param name="phrases">Phrases to use in the vocabulary.</param>
/// <param name="vocabularyName">Name for the vocabulary.</param>
/// <returns>The state of the custom vocabulary.</returns>
public async Task<VocabularyState> CreateCustomVocabulary(LanguageCode
languageCode,
    List<string> phrases, string vocabularyName)
{
    var response = await _amazonTranscribeService.CreateVocabularyAsync(
        new CreateVocabularyRequest
        {
            LanguageCode = languageCode,
            Phrases = phrases,
            VocabularyName = vocabularyName
        });
    return response.VocabularyState;
}
```

- Para obter detalhes da API, consulte [CreateVocabulary](#) em Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como criar um vocabulário personalizado

O exemplo de `create-vocabulary` a seguir cria um vocabulário personalizado. Para criar um vocabulário personalizado, você deve ter criado um arquivo de texto com todos os termos que deseja transcrever com mais precisão. Para `vocabulary-file-uri`, especifique o URI do Amazon Simple Storage Service (Amazon S3) desse arquivo de texto. Em `language-code`, especifique um código de idioma correspondente ao idioma do vocabulário personalizado. Em `vocabulary-name`, especifique como você deseja chamar seu vocabulário personalizado.

```
aws transcribe create-vocabulary \  
  --language-code language-code \  
  --vocabulary-name cli-vocab-example \  
  --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/the-text-file-  
for-the-custom-vocabulary.txt
```

Saída:

```
{  
  "VocabularyName": "cli-vocab-example",  
  "LanguageCode": "language-code",  
  "VocabularyState": "PENDING"  
}
```

Para obter mais informações, consulte [Custom Vocabularies](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [CreateVocabulary](#) em Referência de AWS CLI Comandos.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
def create_vocabulary(
    vocabulary_name, language_code, transcribe_client, phrases=None,
    table_uri=None
):
    """
    Creates a custom vocabulary that can be used to improve the accuracy of
    transcription jobs. This function returns as soon as the vocabulary
    processing
    is started. Call get_vocabulary to get the current status of the vocabulary.
    The vocabulary is ready to use when its status is 'READY'.

    :param vocabulary_name: The name of the custom vocabulary.
    :param language_code: The language code of the vocabulary.
        For example, en-US or nl-NL.
    :param transcribe_client: The Boto3 Transcribe client.
    :param phrases: A list of comma-separated phrases to include in the
    vocabulary.
    :param table_uri: A table of phrases and pronunciation hints to include in
    the
        vocabulary.
    :return: Information about the newly created vocabulary.
    """
    try:
        vocab_args = {"VocabularyName": vocabulary_name, "LanguageCode":
language_code}
        if phrases is not None:
            vocab_args["Phrases"] = phrases
        elif table_uri is not None:
            vocab_args["VocabularyFileUri"] = table_uri
        response = transcribe_client.create_vocabulary(**vocab_args)
        logger.info("Created custom vocabulary %s.", response["VocabularyName"])
    except ClientError:
```

```
        logger.exception("Couldn't create custom vocabulary %s.",
            vocabulary_name)
        raise
    else:
        return response
```

- Para obter detalhes da API, consulte a [CreateVocabulary](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use `DeleteMedicalTranscriptionJob` com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `DeleteMedicalTranscriptionJob`.

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
    /// <summary>
    /// Delete a medical transcription job. Also deletes the transcript
    associated with the job.
    /// </summary>
    /// <param name="jobName">Name of the medical transcription job to delete.</
param>
    /// <returns>True if successful.</returns>
    public async Task<bool> DeleteMedicalTranscriptionJob(string jobName)
    {
```

```
var response = await
_amazonTranscribeService.DeleteMedicalTranscriptionJobAsync(
    new DeleteMedicalTranscriptionJobRequest()
    {
        MedicalTranscriptionJobName = jobName
    });
return response.HttpStatusCode == HttpStatusCode.OK;
}
```

- Para obter detalhes da API, consulte [DeleteMedicalTranscriptionJob](#) Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como excluir um trabalho de transcrição médica

O exemplo de `delete-medical-transcription-job` a seguir exclui um trabalho de transcrição médica.

```
aws transcribe delete-medical-transcription-job \
  --medical-transcription-job-name medical-transcription-job-name
```

Este comando não produz saída.

Para obter mais informações, consulte [DeleteMedicalTranscriptionJob](#) Amazon Transcribe Developer Guide.

- Para obter detalhes da API, consulte [DeleteMedicalTranscriptionJob](#) em Referência de AWS CLI Comandos.

JavaScript

SDK para JavaScript (v3)

Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie o cliente.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

Excluir um trabalho de transcrição médica.

```
// Import the required AWS SDK clients and commands for Node.js
import { DeleteMedicalTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
  MedicalTranscriptionJobName: "MEDICAL_JOB_NAME", // For example,
  'medical_transcription_demo'
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new DeleteMedicalTranscriptionJobCommand(params)
    );
    console.log("Success - deleted");
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
```

```
}  
};  
run();
```

- Para obter mais informações, consulte o [Guia do desenvolvedor do AWS SDK for JavaScript](#).
- Para obter detalhes da API, consulte [DeleteMedicalTranscriptionJob](#) a Referência AWS SDK for JavaScript da API.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **DeleteTranscriptionJob** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `DeleteTranscriptionJob`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação no contexto no seguinte exemplo de código:

- [Criar e refinar um vocabulário personalizado](#)

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>  
/// Delete a transcription job. Also deletes the transcript associated with  
the job.
```

```
/// </summary>
/// <param name="jobName">Name of the transcription job to delete.</param>
/// <returns>True if successful.</returns>
public async Task<bool> DeleteTranscriptionJob(string jobName)
{
    var response = await
    _amazonTranscribeService.DeleteTranscriptionJobAsync(
        new DeleteTranscriptionJobRequest()
        {
            TranscriptionJobName = jobName
        });
    return response.HttpStatusCode == HttpStatusCode.OK;
}
```

- Para obter detalhes da API, consulte [DeleteTranscriptionJob](#) na Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como excluir um dos trabalhos de transcrição

O exemplo de `delete-transcription-job` a seguir exclui um dos trabalhos de transcrição.

```
aws transcribe delete-transcription-job \
    --transcription-job-name your-transcription-job
```

Este comando não produz saída.

Para obter mais informações, consulte [DeleteTranscriptionJob](#) no Amazon Transcribe Developer Guide.

- Para obter detalhes da API, consulte [DeleteTranscriptionJob](#) em Referência de AWS CLI Comandos.

JavaScript

SDK para JavaScript (v3)

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

Excluir um trabalho de transcrição.

```
// Import the required AWS SDK clients and commands for Node.js
import { DeleteTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
  TranscriptionJobName: "JOB_NAME", // Required. For example, 'transcription_demo'
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new DeleteTranscriptionJobCommand(params)
    );
    console.log("Success - deleted");
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Crie o cliente.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
```

```
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

- Para obter mais informações, consulte o [Guia do desenvolvedor do AWS SDK for JavaScript](#).
- Para obter detalhes da API, consulte [DeleteTranscriptionJob](#) a Referência AWS SDK for JavaScript da API.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
def delete_job(job_name, transcribe_client):
    """
    Deletes a transcription job. This also deletes the transcript associated with
    the job.

    :param job_name: The name of the job to delete.
    :param transcribe_client: The Boto3 Transcribe client.
    """
    try:
        transcribe_client.delete_transcription_job(TranscriptionJobName=job_name)
        logger.info("Deleted job %s.", job_name)
    except ClientError:
        logger.exception("Couldn't delete job %s.", job_name)
        raise
```

- Para obter detalhes da API, consulte a [DeleteTranscriptionJob](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **DeleteVocabulary** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `DeleteVocabulary`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação no contexto no seguinte exemplo de código:

- [Criar e refinar um vocabulário personalizado](#)

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// Delete an existing custom vocabulary.
/// </summary>
/// <param name="vocabularyName">Name of the vocabulary to delete.</param>
/// <returns>True if successful.</returns>
public async Task<bool> DeleteCustomVocabulary(string vocabularyName)
{
    var response = await _amazonTranscribeService.DeleteVocabularyAsync(
        new DeleteVocabularyRequest
        {
            VocabularyName = vocabularyName
        });
    return response.HttpStatusCode == HttpStatusCode.OK;
}
```

- Para obter detalhes da API, consulte [DeleteVocabulary](#) em Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como excluir um vocabulário personalizado

O exemplo de `delete-vocabulary` a seguir exclui um vocabulário personalizado.

```
aws transcribe delete-vocabulary \  
  --vocabulary-name vocabulary-name
```

Este comando não produz saída.

Para obter mais informações, consulte [Custom Vocabularies](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [DeleteVocabulary](#) em Referência de AWS CLI Comandos.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
def delete_vocabulary(vocabulary_name, transcribe_client):  
    """  
    Deletes a custom vocabulary.  
  
    :param vocabulary_name: The name of the vocabulary to delete.  
    :param transcribe_client: The Boto3 Transcribe client.  
    """  
    try:  
        transcribe_client.delete_vocabulary(VocabularyName=vocabulary_name)
```

```
logger.info("Deleted vocabulary %s.", vocabulary_name)
except ClientError:
    logger.exception("Couldn't delete vocabulary %s.", vocabulary_name)
    raise
```

- Para obter detalhes da API, consulte a [DeleteVocabulary](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **GetTranscriptionJob** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `GetTranscriptionJob`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação em contexto nos seguintes exemplos de código:

- [Criar e refinar um vocabulário personalizado](#)
- [Transcrever áudio e obter dados do trabalho](#)

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// Get details about a transcription job.
/// </summary>
/// <param name="jobName">A unique name for the transcription job.</param>
```

```
/// <returns>A TranscriptionJob instance with information on the requested
job.</returns>
public async Task<TranscriptionJob> GetTranscriptionJob(string jobName)
{
    var response = await _amazonTranscribeService.GetTranscriptionJobAsync(
        new GetTranscriptionJobRequest()
        {
            TranscriptionJobName = jobName
        });
    return response.TranscriptionJob;
}
```

- Para obter detalhes da API, consulte [GetTranscriptionJob](#) Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como obter informações sobre um trabalho de transcrição específico

O exemplo de `get-transcription-job` a seguir coleta informações sobre um trabalho de transcrição específico. Para acessar os resultados da transcrição, use o `TranscriptFileUri` parâmetro. Use o `MediaFileUri` parâmetro para ver qual arquivo de áudio você transcreveu com esse trabalho. É possível usar o objeto `Configurações` para ver os recursos opcionais habilitados no trabalho de transcrição.

```
aws transcribe get-transcription-job \
  --transcription-job-name your-transcription-job
```

Saída:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "your-transcription-job",
    "TranscriptionJobStatus": "COMPLETED",
    "LanguageCode": "language-code",
    "MediaSampleRateHertz": 48000,
    "MediaFormat": "mp4",
    "Media": {
```

```
        "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-
extension"
    },
    "Transcript": {
        "TranscriptFileUri": "https://Amazon-S3-file-location-of-
transcription-output"
    },
    "StartTime": "2020-09-18T22:27:23.970000+00:00",
    "CreationTime": "2020-09-18T22:27:23.948000+00:00",
    "CompletionTime": "2020-09-18T22:28:21.197000+00:00",
    "Settings": {
        "ChannelIdentification": false,
        "ShowAlternatives": false
    },
    "IdentifyLanguage": true,
    "IdentifiedLanguageScore": 0.8672199249267578
}
}
```

Para obter mais informações, consulte [Conceitos básicos \(interface de linha de AWS comando\)](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [GetTranscriptionJob](#) em Referência de AWS CLI Comandos.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
def get_job(job_name, transcribe_client):
    """
    Gets details about a transcription job.

    :param job_name: The name of the job to retrieve.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: The retrieved transcription job.
```

```
"""
try:
    response = transcribe_client.get_transcription_job(
        TranscriptionJobName=job_name
    )
    job = response["TranscriptionJob"]
    logger.info("Got job %s.", job["TranscriptionJobName"])
except ClientError:
    logger.exception("Couldn't get job %s.", job_name)
    raise
else:
    return job
```

- Para obter detalhes da API, consulte a [GetTranscriptionJob](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **GetVocabulary** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar GetVocabulary.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação no contexto no seguinte exemplo de código:

- [Criar e refinar um vocabulário personalizado](#)

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).


```
/// <summary>
/// Get information about a custom vocabulary.
/// </summary>
/// <param name="vocabularyName">Name of the vocabulary.</param>
/// <returns>The state of the custom vocabulary.</returns>
public async Task<VocabularyState> GetCustomVocabulary(string vocabularyName)
{
    var response = await _amazonTranscribeService.GetVocabularyAsync(
        new GetVocabularyRequest()
        {
            VocabularyName = vocabularyName
        });
    return response.VocabularyState;
}
```

- Para obter detalhes da API, consulte [GetVocabulary](#) a Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como obter informações sobre um vocabulário personalizado

O exemplo de `get-vocabulary` a seguir coleta informações sobre um vocabulário personalizado criado anteriormente.

```
aws transcribe get-vocabulary \
  --vocabulary-name cli-vocab-1
```

Saída:

```
{
  "VocabularyName": "cli-vocab-1",
  "LanguageCode": "language-code",
  "VocabularyState": "READY",
  "LastModifiedTime": "2020-09-19T23:22:32.836000+00:00",
  "DownloadUri": "https://link-to-download-the-text-file-used-to-create-your-
  custom-vocabulary"
```

```
}
```

Para obter mais informações, consulte [Custom Vocabularies](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [GetVocabulary](#) em Referência de AWS CLI Comandos.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
def get_vocabulary(vocabulary_name, transcribe_client):
    """
    Gets information about a custom vocabulary.

    :param vocabulary_name: The name of the vocabulary to retrieve.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: Information about the vocabulary.
    """
    try:
        response =
transcribe_client.get_vocabulary(VocabularyName=vocabulary_name)
        logger.info("Got vocabulary %s.", response["VocabularyName"])
    except ClientError:
        logger.exception("Couldn't get vocabulary %s.", vocabulary_name)
        raise
    else:
        return response
```

- Para obter detalhes da API, consulte a [GetVocabulary](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use `ListMedicalTranscriptionJobs` com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `ListMedicalTranscriptionJobs`.

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// List medical transcription jobs, optionally with a name filter.
/// </summary>
/// <param name="jobNameContains">Optional name filter for the medical
transcription jobs.</param>
/// <returns>A list of summaries about medical transcription jobs.</returns>
public async Task<List<MedicalTranscriptionJobSummary>>
ListMedicalTranscriptionJobs(
    string? jobNameContains = null)
{
    var response = await
_amazonTranscribeService.ListMedicalTranscriptionJobsAsync(
    new ListMedicalTranscriptionJobsRequest()
    {
        JobNameContains = jobNameContains
    });
    return response.MedicalTranscriptionJobSummaries;
}
```

- Para obter detalhes da API, consulte [ListMedicalTranscriptionJobs](#) Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como listar os trabalhos de transcrição médica

O `list-medical-transcription-jobs` exemplo a seguir lista os trabalhos de transcrição médica associados à sua AWS conta e região. Para obter mais informações sobre um trabalho de transcrição específico, copie o valor de um `MedicalTranscriptionJobName` parâmetro na saída da transcrição e especifique esse valor para a `MedicalTranscriptionJobName` opção do `get-medical-transcription-job` comando. Para ver mais trabalhos de transcrição, copie o valor do `NextToken` parâmetro, execute o `list-medical-transcription-jobs` comando novamente e especifique esse valor na `--next-token` opção.

```
aws transcribe list-medical-transcription-jobs
```

Saída:

```
{
  "NextToken": "3/PblzkiGhzjER3KHuQt2fmbPLF7cDYafjFMEoGn440N/
gsuUSTIkGyanvRE6WMXFd/ZTEc2EZj+P9eii/
z102FDYli6RLI0WoRX4RwMisVrh9G0Kie0Y8ikBCdtqLZB10Wa9McC+eb0l
+LaDtZPC4u6ttoHLRL1EfzqstHXSgapXg3tEBtm9piIaPB6M0M5BB6t86+qtmocTR/
qrteHZBBudhTfbCwhsxaqujHiiUvFdm3BQbKKWIW06yV9b+4f38oD2lVIan
+vfUs3gBYA15VTDmXXzQPBQ0HPjtwmFI+IWX15nSUjWuN3TUylHgPWzDaYT8qBtu0Z+3UG4V6b
+K2CC0XszXg5rBq9hYgNzy4XoFh/6s5DoSnzq49Q9xHgHdT2yBADFmvFK7myZBsJ75+2vQZ0SVpWUPy3WT/32zFAc
+mFYfUjtTZ8n/jq7aQEjQ42A
+X/7K6Jg0cdVPtEg8P1Dr5kgYYG3q30mYXX37U3FZuJmnTI63VtIXsNn0U5eGoY0btpk00Nq9UkzgSJxqj84ZD5n
+S0EGy9ZUYBJRRcGeYUM3Q4DbSJfUwSAqcFdLIWZdp8qIREMQIBWy7BLwSdyqsQo2vRrd53hm5aWM7SVf6pPq6X/
IXR5+1eU00D8/coaTT4ES2DerbV6RkV4o0VT1d0SdVX/
MmtkNG8nYj8PqU07w7988quh1ZP6D80veJS1q73tUUR9MjnGernW2tAnvnLNhdefBcD
+sZVfYq3iBMFY7wTy1P1G6NqW9GrYDYox3tTPW1D7phpbVSyKrh/
PdYrps5UxnsGoA1b7L/FfAXDfUoGrGUB4N3JsPYXX9D++g+6gV1qBBs/
Wff934aKqfD6UTggm/zV3GA0WiBpfvAZRvEb924i6yGHyMC7y5401ZAwSBupmI
+FFd13CaP04kN1vJlth6aM5vUPXg4BpyUhtbRhwD/KxCvf9K0tLJGyL1A==",
  "MedicalTranscriptionJobSummaries": [
    {
```

```
    "MedicalTranscriptionJobName": "vocabulary-dictation-medical-
transcription-job",
    "CreationTime": "2020-09-21T21:17:27.016000+00:00",
    "StartTime": "2020-09-21T21:17:27.045000+00:00",
    "CompletionTime": "2020-09-21T21:17:59.561000+00:00",
    "LanguageCode": "en-US",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "CUSTOMER_BUCKET",
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  },
  {
    "MedicalTranscriptionJobName": "alternatives-dictation-medical-
transcription-job",
    "CreationTime": "2020-09-21T21:01:14.569000+00:00",
    "StartTime": "2020-09-21T21:01:14.592000+00:00",
    "CompletionTime": "2020-09-21T21:01:43.606000+00:00",
    "LanguageCode": "en-US",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "CUSTOMER_BUCKET",
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  },
  {
    "MedicalTranscriptionJobName": "alternatives-conversation-medical-
transcription-job",
    "CreationTime": "2020-09-21T19:09:18.171000+00:00",
    "StartTime": "2020-09-21T19:09:18.199000+00:00",
    "CompletionTime": "2020-09-21T19:10:22.516000+00:00",
    "LanguageCode": "en-US",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "CUSTOMER_BUCKET",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  },
  {
    "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
    "CreationTime": "2020-09-21T18:43:37.157000+00:00",
    "StartTime": "2020-09-21T18:43:37.265000+00:00",
    "CompletionTime": "2020-09-21T18:44:21.192000+00:00",
    "LanguageCode": "en-US",
    "TranscriptionJobStatus": "COMPLETED",
    "OutputLocationType": "CUSTOMER_BUCKET",
```

```
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    },
    {
        "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
        "CreationTime": "2020-09-20T23:46:44.053000+00:00",
        "StartTime": "2020-09-20T23:46:44.081000+00:00",
        "CompletionTime": "2020-09-20T23:47:35.851000+00:00",
        "LanguageCode": "en-US",
        "TranscriptionJobStatus": "COMPLETED",
        "OutputLocationType": "CUSTOMER_BUCKET",
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    }
]
}
```

Para obter mais informações, consulte <https://docs.aws.amazon.com/transcribe/latest/dg/batch-med-transcription.html> no Amazon Transcribe Developer Guide.

- Para obter detalhes da API, consulte [ListMedicalTranscriptionJobs](#) em Referência de AWS CLI Comandos.

JavaScript

SDK para JavaScript (v3)

Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie o cliente.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
```

```
export { transcribeClient };
```

Listar trabalhos de transcrição médica.

```
// Import the required AWS SDK clients and commands for Node.js
import { StartMedicalTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
  MedicalTranscriptionJobName: "MEDICAL_JOB_NAME", // Required
  OutputBucketName: "OUTPUT_BUCKET_NAME", // Required
  Specialty: "PRIMARYCARE", // Required. Possible values are 'PRIMARYCARE'
  Type: "JOB_TYPE", // Required. Possible values are 'CONVERSATION' and
  'DICTATION'
  LanguageCode: "LANGUAGE_CODE", // For example, 'en-US'
  MediaFormat: "SOURCE_FILE_FORMAT", // For example, 'wav'
  Media: {
    MediaFileUri: "SOURCE_FILE_LOCATION",
    // The S3 object location of the input media file. The URI must be in the
    same region
    // as the API endpoint that you are calling. For example,
    // "https://transcribe-demo.s3-REGION.amazonaws.com/hello_world.wav"
  },
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new StartMedicalTranscriptionJobCommand(params)
    );
    console.log("Success - put", data);
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};

run();
```

- Para obter mais informações, consulte o [Guia do desenvolvedor do AWS SDK for JavaScript](#).

- Para obter detalhes da API, consulte [ListMedicalTranscriptionJobs](#) na Referência AWS SDK for JavaScript da API.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **ListTranscriptionJobs** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `ListTranscriptionJobs`.

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// List transcription jobs, optionally with a name filter.
/// </summary>
/// <param name="jobNameContains">Optional name filter for the transcription
jobs.</param>
/// <returns>A list of transcription job summaries.</returns>
public async Task<List<TranscriptionJobSummary>>
ListTranscriptionJobs(string? jobNameContains = null)
{
    var response = await _amazonTranscribeService.ListTranscriptionJobsAsync(
        new ListTranscriptionJobsRequest()
        {
            JobNameContains = jobNameContains
        });
    return response.TranscriptionJobSummaries;
}
```


- Para obter detalhes da API, consulte [ListTranscriptionJobs](#) na Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como listar os trabalhos de transcrição

O `list-transcription-jobs` exemplo a seguir lista os trabalhos de transcrição associados à sua AWS conta e região.

```
aws transcribe list-transcription-jobs
```

Saída:

```
{
  "NextToken": "NextToken",
  "TranscriptionJobSummaries": [
    {
      "TranscriptionJobName": "speak-id-job-1",
      "CreationTime": "2020-08-17T21:06:15.391000+00:00",
      "StartTime": "2020-08-17T21:06:15.416000+00:00",
      "CompletionTime": "2020-08-17T21:07:05.098000+00:00",
      "LanguageCode": "language-code",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "SERVICE_BUCKET"
    },
    {
      "TranscriptionJobName": "job-1",
      "CreationTime": "2020-08-17T20:50:24.207000+00:00",
      "StartTime": "2020-08-17T20:50:24.230000+00:00",
      "CompletionTime": "2020-08-17T20:52:18.737000+00:00",
      "LanguageCode": "language-code",
      "TranscriptionJobStatus": "COMPLETED",
      "OutputLocationType": "SERVICE_BUCKET"
    },
    {
      "TranscriptionJobName": "sdk-test-job-4",
      "CreationTime": "2020-08-17T20:32:27.917000+00:00",
      "StartTime": "2020-08-17T20:32:27.956000+00:00",
      "CompletionTime": "2020-08-17T20:33:15.126000+00:00",
```

```

        "LanguageCode": "language-code",
        "TranscriptionJobStatus": "COMPLETED",
        "OutputLocationType": "SERVICE_BUCKET"
    },
    {
        "TranscriptionJobName": "Diarization-speak-id",
        "CreationTime": "2020-08-10T22:10:09.066000+00:00",
        "StartTime": "2020-08-10T22:10:09.116000+00:00",
        "CompletionTime": "2020-08-10T22:26:48.172000+00:00",
        "LanguageCode": "language-code",
        "TranscriptionJobStatus": "COMPLETED",
        "OutputLocationType": "SERVICE_BUCKET"
    },
    {
        "TranscriptionJobName": "your-transcription-job-name",
        "CreationTime": "2020-07-29T17:45:09.791000+00:00",
        "StartTime": "2020-07-29T17:45:09.826000+00:00",
        "CompletionTime": "2020-07-29T17:46:20.831000+00:00",
        "LanguageCode": "language-code",
        "TranscriptionJobStatus": "COMPLETED",
        "OutputLocationType": "SERVICE_BUCKET"
    }
]
}

```

Para obter mais informações, consulte [Conceitos básicos \(interface de linha de AWS comando\)](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [ListTranscriptionJobs](#) em Referência de AWS CLI Comandos.

Java

SDK para Java 2.x

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
public class ListTranscriptionJobs {
```

```
public static void main(String[] args) {
    TranscribeClient transcribeClient = TranscribeClient.builder()
        .region(Region.US_EAST_1)
        .build();

    listTranscriptionJobs(transcribeClient);
}

public static void listTranscriptionJobs(TranscribeClient
transcribeClient) {
    ListTranscriptionJobsRequest listJobsRequest =
ListTranscriptionJobsRequest.builder()
        .build();

transcribeClient.listTranscriptionJobsPaginator(listJobsRequest).stream()
    .flatMap(response ->
response.transcriptionJobSummaries().stream())
    .forEach(jobSummary -> {
        System.out.println("Job Name: " +
jobSummary.transcriptionJobName());
        System.out.println("Job Status: " +
jobSummary.transcriptionJobStatus());
        System.out.println("Output Location: " +
jobSummary.outputLocationType());
        // Add more information as needed

        // Retrieve additional details for the job if necessary
        GetTranscriptionJobResponse jobDetails =
transcribeClient.getTranscriptionJob(
            GetTranscriptionJobRequest.builder()

.transcriptionJobName(jobSummary.transcriptionJobName())
                .build());

        // Display additional details
        System.out.println("Language Code: " +
jobDetails.transcriptionJob().languageCode());
        System.out.println("Media Format: " +
jobDetails.transcriptionJob().mediaFormat());
        // Add more details as needed

        System.out.println("-----");
    });
}
```

```
    }  
  }  
}
```

- Para obter detalhes da API, consulte [ListTranscriptionJobs](#) na Referência AWS SDK for Java 2.x da API.

JavaScript

SDK para JavaScript (v3)

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

Listar trabalhos de transcrição.

```
// Import the required AWS SDK clients and commands for Node.js  
  
import { ListTranscriptionJobsCommand } from "@aws-sdk/client-transcribe";  
import { transcribeClient } from "../libs/transcribeClient.js";  
  
// Set the parameters  
export const params = {  
  JobNameContains: "KEYWORD", // Not required. Returns only transcription  
  // job names containing this string  
};  
  
export const run = async () => {  
  try {  
    const data = await transcribeClient.send(  
      new ListTranscriptionJobsCommand(params)  
    );  
    console.log("Success", data.TranscriptionJobSummaries);  
    return data; // For unit tests.  
  } catch (err) {  
    console.log("Error", err);  
  }  
};  
run();
```


Crie o cliente.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

- Para obter mais informações, consulte o [Guia do desenvolvedor do AWS SDK for JavaScript](#).
- Para obter detalhes da API, consulte [ListTranscriptionJobs](#) Referência AWS SDK for JavaScript da API.

Python

SDK para Python (Boto3).

 Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
def list_jobs(job_filter, transcribe_client):
    """
    Lists summaries of the transcription jobs for the current AWS account.

    :param job_filter: The list of returned jobs must contain this string in
    their
                       names.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: The list of retrieved transcription job summaries.
    """
    try:
        response =
        transcribe_client.list_transcription_jobs(JobNameContains=job_filter)
```

```
jobs = response["TranscriptionJobSummaries"]
next_token = response.get("NextToken")
while next_token is not None:
    response = transcribe_client.list_transcription_jobs(
        JobNameContains=job_filter, NextToken=next_token
    )
    jobs += response["TranscriptionJobSummaries"]
    next_token = response.get("NextToken")
logger.info("Got %s jobs with filter %s.", len(jobs), job_filter)
except ClientError:
    logger.exception("Couldn't get jobs with filter %s.", job_filter)
    raise
else:
    return jobs
```

- Para obter detalhes da API, consulte a [ListTranscriptionJobs](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **ListVocabularies** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `ListVocabularies`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação no contexto no seguinte exemplo de código:

- [Criar e refinar um vocabulário personalizado](#)

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// List custom vocabularies for the current account. Optionally specify a
name
/// filter and a specific state to filter the vocabularies list.
/// </summary>
/// <param name="nameContains">Optional string the vocabulary name must
contain.</param>
/// <param name="stateEquals">Optional state of the vocabulary.</param>
/// <returns>List of information about the vocabularies.</returns>
public async Task<List<VocabularyInfo>> ListCustomVocabularies(string?
nameContains = null,
    VocabularyState? stateEquals = null)
{
    var response = await _amazonTranscribeService.ListVocabulariesAsync(
        new ListVocabulariesRequest()
        {
            NameContains = nameContains,
            StateEquals = stateEquals
        });
    return response.Vocabularies;
}
```

- Para obter detalhes da API, consulte [ListVocabularies](#) a Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como listar os vocabulários personalizados

O `list-vocabularies` exemplo a seguir lista os vocabulários personalizados associados à sua AWS conta e região.

```
aws transcribe list-vocabularies
```

Saída:

```
{
  "NextToken": "NextToken",
  "Vocabularies": [
    {
      "VocabularyName": "ards-test-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-27T22:00:27.330000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "sample-test",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T23:04:11.044000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-test-3-1",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T22:12:22.277000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-test-2",
      "LanguageCode": "language-code",
      "LastModifiedTime": "2020-04-24T21:53:50.455000+00:00",
      "VocabularyState": "READY"
    },
    {
      "VocabularyName": "CRLF-to-LF-1-1",
      "LanguageCode": "language-code",

```



```

        "LastModifiedTime": "2020-04-24T21:39:33.356000+00:00",
        "VocabularyState": "READY"
    }
]
}

```

Para obter mais informações, consulte [Custom Vocabularies](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [ListVocabularies](#) em Referência de AWS CLI Comandos.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```

def list_vocabularies(vocabulary_filter, transcribe_client):
    """
    Lists the custom vocabularies created for this AWS account.

    :param vocabulary_filter: The returned vocabularies must contain this string
    in
                               their names.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: The list of retrieved vocabularies.
    """
    try:
        response =
transcribe_client.list_vocabularies(NameContains=vocabulary_filter)
        vocabs = response["Vocabularies"]
        next_token = response.get("NextToken")
        while next_token is not None:
            response = transcribe_client.list_vocabularies(
                NameContains=vocabulary_filter, NextToken=next_token
            )
            vocabs += response["Vocabularies"]

```

```
        next_token = response.get("NextToken")
    logger.info(
        "Got %s vocabularies with filter %s.", len(vocabs), vocabulary_filter
    )
except ClientError:
    logger.exception(
        "Couldn't list vocabularies with filter %s.", vocabulary_filter
    )
    raise
else:
    return vocabs
```

- Para obter detalhes da API, consulte a [ListVocabularies](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **StartMedicalTranscriptionJob** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `StartMedicalTranscriptionJob`.

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// Start a medical transcription job for a media file. This method returns
/// as soon as the job is started.
/// </summary>
```

```

    /// <param name="jobName">A unique name for the medical transcription job.</
param>
    /// <param name="mediaFileUri">The URI of the media file, typically an Amazon
S3 location.</param>
    /// <param name="mediaFormat">The format of the media file.</param>
    /// <param name="outputBucketName">Location for the output, typically an
Amazon S3 location.</param>
    /// <param name="transcriptionType">Conversation or dictation transcription
type.</param>
    /// <returns>A MedicalTransactionJob instance with information on the new
job.</returns>
    public async Task<MedicalTranscriptionJob> StartMedicalTranscriptionJob(
        string jobName, string mediaFileUri,
        MediaFormat mediaFormat, string outputBucketName,
        Amazon.TranscribeService.Type transcriptionType)
    {
        var response = await
        _amazonTranscribeService.StartMedicalTranscriptionJobAsync(
            new StartMedicalTranscriptionJobRequest()
            {
                MedicalTranscriptionJobName = jobName,
                Media = new Media()
                {
                    MediaFileUri = mediaFileUri
                },
                MediaFormat = mediaFormat,
                LanguageCode =
                    LanguageCode
                    .EnUS, // The value must be en-US for medical
transcriptions.
                OutputBucketName = outputBucketName,
                OutputKey =
                    jobName, // The value is a key used to fetch the output of
the transcription.
                Specialty = Specialty.PRIMARYCARE, // The value PRIMARYCARE must
be set.
                Type = transcriptionType
            });
        return response.MedicalTranscriptionJob;
    }

```

- Para obter detalhes da API, consulte [StartMedicalTranscriptionJob](#) Referência AWS SDK for .NET da API.

CLI

AWS CLI

Exemplo 1: como transcrever um ditado médico armazenado como um arquivo de áudio

O exemplo de `start-medical-transcription-job` a seguir transcreve um arquivo de áudio. Especifique o local da saída da transcrição no parâmetro `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://myfile.json
```

Conteúdo de `myfile.json`:

```
{  
  "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "DICTATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  }  
}
```

Saída:

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-  
job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
    },  
    "StartTime": "2020-09-20T00:35:22.256000+00:00",  
    "CreationTime": "2020-09-20T00:35:22.218000+00:00",  
  }  
}
```

```
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}
```

Para obter mais informações, consulte [Batch Transcription Overview](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 2: como transcrever um diálogo entre médico e paciente armazenado como um arquivo de áudio

O exemplo de `start-medical-transcription-job` a seguir transcreve um arquivo de áudio contendo um diálogo entre médico e paciente. Você especifica a localização da saída da transcrição no `OutputBucketName` parâmetro.

```
aws transcribe start-medical-transcription-job \
  --cli-input-json file://mysecondfile.json
```

Conteúdo de `mysecondfile.json`:

```
{
  "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  }
}
```

Saída:

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "simple-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
  },
}
```

```

    "StartTime": "2020-09-20T23:19:49.965000+00:00",
    "CreationTime": "2020-09-20T23:19:49.941000+00:00",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}

```

Para obter mais informações, consulte [Batch Transcription Overview](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 3: como transcrever um arquivo de áudio multicanal de um diálogo entre médico e paciente

O exemplo de `start-medical-transcription-job` a seguir transcreve o áudio de cada canal no arquivo de áudio e mescla as transcrições separadas de cada canal em uma única saída de transcrição. Especifique o local da saída da transcrição no parâmetro `OutputBucketName`.

```

aws transcribe start-medical-transcription-job \
  --cli-input-json file://mythirdfile.json

```

Conteúdo de `mythirdfile.json`:

```

{
  "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  },
  "Settings": {
    "ChannelIdentification": true
  }
}

```

Saída:

```

{

```

```

    "MedicalTranscriptionJob": {
      "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
      "TranscriptionJobStatus": "IN_PROGRESS",
      "LanguageCode": "language-code",
      "Media": {
        "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
      },
      "StartTime": "2020-09-20T23:46:44.081000+00:00",
      "CreationTime": "2020-09-20T23:46:44.053000+00:00",
      "Settings": {
        "ChannelIdentification": true
      },
      "Specialty": "PRIMARYCARE",
      "Type": "CONVERSATION"
    }
  }
}

```

Para obter mais informações, consulte [Channel Identification](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 4: como transcrever um arquivo de áudio de um diálogo entre médico e paciente e identificar quem está falando na saída da transcrição

O exemplo de `start-medical-transcription-job` a seguir transcreve um arquivo de áudio e rotula a fala de cada locutor na saída da transcrição. Especifique o local da saída da transcrição no parâmetro `OutputBucketName`.

```

aws transcribe start-medical-transcription-job \
  --cli-input-json file://myfourthfile.json

```

Conteúdo de `myfourthfile.json`:

```

{
  "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  }
}

```

```

    },
    "Settings":{
      "ShowSpeakerLabels": true,
      "MaxSpeakerLabels": 2
    }
  }
}

```

Saída:

```

{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T18:43:37.265000+00:00",
    "CreationTime": "2020-09-21T18:43:37.157000+00:00",
    "Settings": {
      "ShowSpeakerLabels": true,
      "MaxSpeakerLabels": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION"
  }
}

```

Para obter mais informações, consulte [Identifying Speakers](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 5: como transcrever uma conversa médica armazenada como um arquivo de áudio com até duas alternativas de transcrição

O exemplo de `start-medical-transcription-job` a seguir cria até duas transcrições alternativas de um único arquivo de áudio. Cada transcrição tem um nível de confiança associado a ela. Por padrão, o Amazon Transcribe retorna a transcrição com o maior nível de confiança. É possível especificar que o Amazon Transcribe retorne transcrições adicionais com níveis de confiança mais baixos. Especifique o local da saída da transcrição no parâmetro `OutputBucketName`.


```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://myfifthfile.json
```

Conteúdo de myfifthfile.json:

```
{  
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-  
transcription-job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  },  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 2  
  }  
}
```

Saída:

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "alternatives-conversation-medical-  
transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
    },  
    "StartTime": "2020-09-21T19:09:18.199000+00:00",  
    "CreationTime": "2020-09-21T19:09:18.171000+00:00",  
    "Settings": {  
      "ShowAlternatives": true,  
      "MaxAlternatives": 2  
    },  
    "Specialty": "PRIMARYCARE",  
    "Type": "CONVERSATION"  
  }  
}
```

Para obter mais informações, consulte [Alternative Transcriptions](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 6: como transcrever um arquivo de áudio de um ditado médico com até duas transcrições alternativas

O exemplo de `start-medical-transcription-job` a seguir transcreve um arquivo de áudio e usa um filtro de vocabulário para mascarar qualquer palavra indesejada. Você especifica a localização da saída da transcrição no `OutputBucketName` parâmetro.

```
aws transcribe start-medical-transcription-job \  
  --cli-input-json file://mysixthfile.json
```

Conteúdo de `mysixthfile.json`:

```
{  
  "MedicalTranscriptionJobName": "alternatives-conversation-medical-  
transcription-job",  
  "LanguageCode": "language-code",  
  "Specialty": "PRIMARYCARE",  
  "Type": "DICTATION",  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
  },  
  "Settings": {  
    "ShowAlternatives": true,  
    "MaxAlternatives": 2  
  }  
}
```

Saída:

```
{  
  "MedicalTranscriptionJob": {  
    "MedicalTranscriptionJobName": "alternatives-dictation-medical-  
transcription-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "language-code",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"  
    },  
  },  
}
```

```

    "StartTime": "2020-09-21T21:01:14.592000+00:00",
    "CreationTime": "2020-09-21T21:01:14.569000+00:00",
    "Settings": {
      "ShowAlternatives": true,
      "MaxAlternatives": 2
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}

```

Para obter mais informações, consulte [Alternative Transcriptions](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 7: como transcrever um arquivo de áudio de um ditado médico com maior precisão usando um vocabulário personalizado

O exemplo de `start-medical-transcription-job` a seguir transcreve um arquivo de áudio e usa um vocabulário médico personalizado que você criou anteriormente para aumentar a precisão da transcrição. Especifique o local da saída da transcrição no parâmetro `OutputBucketName`.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myseventhfile.json

```

Conteúdo de `mysixthfile.json`:

```

{
  "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-
job",
  "LanguageCode": "language-code",
  "Specialty": "PRIMARYCARE",
  "Type": "DICTATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
  },
  "Settings": {
    "VocabularyName": "cli-medical-vocab-1"
  }
}

```

Saída:

```
{
  "MedicalTranscriptionJob": {
    "MedicalTranscriptionJobName": "vocabulary-dictation-medical-
transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.extension"
    },
    "StartTime": "2020-09-21T21:17:27.045000+00:00",
    "CreationTime": "2020-09-21T21:17:27.016000+00:00",
    "Settings": {
      "VocabularyName": "cli-medical-vocab-1"
    },
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION"
  }
}
```

Para obter mais informações, consulte [Medical Custom Vocabularies](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [StartMedicalTranscriptionJob](#) em Referência de AWS CLI Comandos.

JavaScript

SDK para JavaScript (v3)

Note

Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no [AWS Code Examples Repository](#).

Crie o cliente.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
```

```
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

Iniciar um trabalho de transcrição médica.

```
// Import the required AWS SDK clients and commands for Node.js
import { StartMedicalTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
  MedicalTranscriptionJobName: "MEDICAL_JOB_NAME", // Required
  OutputBucketName: "OUTPUT_BUCKET_NAME", // Required
  Specialty: "PRIMARYCARE", // Required. Possible values are 'PRIMARYCARE'
  Type: "JOB_TYPE", // Required. Possible values are 'CONVERSATION' and
  'DICTATION'
  LanguageCode: "LANGUAGE_CODE", // For example, 'en-US'
  MediaFormat: "SOURCE_FILE_FORMAT", // For example, 'wav'
  Media: {
    MediaFileUri: "SOURCE_FILE_LOCATION",
    // The S3 object location of the input media file. The URI must be in the
    same region
    // as the API endpoint that you are calling. For example,
    // "https://transcribe-demo.s3-REGION.amazonaws.com/hello_world.wav"
  },
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new StartMedicalTranscriptionJobCommand(params)
    );
    console.log("Success - put", data);
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

- Para obter mais informações, consulte o [Guia do desenvolvedor do AWS SDK for JavaScript](#).
- Para obter detalhes da API, consulte [StartMedicalTranscriptionJob](#) Referência AWS SDK for JavaScript da API.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use `StartStreamTranscriptionAsync` com um AWS SDK ou CLI

O código de exemplo a seguir mostra como usar `StartStreamTranscriptionAsync`.

C++

SDK para C++

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
int main() {
    Aws::SDKOptions options;

    Aws::InitAPI(options);
    {
        //TODO(User): Set to the region of your AWS account.
        const Aws::String region = Aws::Region::US_WEST_2;

        //Load a profile that has been granted AmazonTranscribeFullAccess AWS
        managed permission policy.
        Aws::Client::ClientConfiguration config;
#ifdef _WIN32
        // ATTENTION: On Windows with the AWS C++ SDK, this example only runs if
        the SDK is built
        // with the curl library.
        // For more information, see the accompanying ReadMe.
        // For more information, see "Building the SDK for Windows with curl".
```

```

// https://docs.aws.amazon.com/sdk-for-cpp/v1/developer-guide/setup-
windows.html
//TODO(User): Update to the location of your .crt file.
config.caFile = "C:/curl/bin/curl-ca-bundle.crt";
#endif

config.region = region;

TranscribeStreamingServiceClient client(config);
StartStreamTranscriptionHandler handler;
handler.SetOnErrorCallback(
    [](const Aws::Client::AWSError<TranscribeStreamingServiceErrors>
&error) {
        std::cerr << "ERROR: " + error.GetMessage() << std::endl;
    });
//SetTranscriptEventCallback called for every 'chunk' of file
transcribed.
// Partial results are returned in real time.
handler.SetTranscriptEventCallback([](const TranscriptEvent &ev) {
    for (auto &r: ev.GetTranscript().GetResults()) {
        if (r.GetIsPartial()) {
            std::cout << "[partial] ";
        }
        else {
            std::cout << "[Final] ";
        }
        for (auto &alt: r.GetAlternatives()) {
            std::cout << alt.GetTranscript() << std::endl;
        }
    }
});

StartStreamTranscriptionRequest request;
request.SetMediaSampleRateHertz(SAMPLE_RATE);
request.SetLanguageCode(LanguageCode::en_US);
request.SetMediaEncoding(
    MediaEncoding::pcm); // wav and aiff files are PCM formats.
request.SetEventStreamHandler(handler);

auto OnStreamReady = [](AudioStream &stream) {
    Aws::FStream file(FILE_NAME, std::ios_base::in |
std::ios_base::binary);
    if (!file.is_open()) {
        std::cerr << "Failed to open " << FILE_NAME << '\n';
    }
}

```

```
std::array<char, BUFFER_SIZE> buf;
int i = 0;
while (file) {
    file.read(&buf[0], buf.size());

    if (!file)
        std::cout << "File: only " << file.gcount() << " could be
read"
                << std::endl;

    Aws::Vector<unsigned char> bits{buf.begin(), buf.end()};
    AudioEvent event(std::move(bits));
    if (!stream) {
        std::cerr << "Failed to create a stream" << std::endl;
        break;
    }
    //The std::basic_istream::gcount() is used to count the
characters in the given string. It returns
//the number of characters extracted by the last read()
operation.

    if (file.gcount() > 0) {
        if (!stream.WriteAudioEvent(event)) {
            std::cerr << "Failed to write an audio event" <<
std::endl;

            break;
        }
    }
    else {
        break;
    }
    std::this_thread::sleep_for(std::chrono::milliseconds(
file.
    25)); // Slow down because we are streaming from a

}
if (!stream.WriteAudioEvent(
    AudioEvent())) {
    // Per the spec, we have to send an empty event (an event
without a payload) at the end.
    std::cerr << "Failed to send an empty frame" << std::endl;
}
else {
    std::cout << "Successfully sent the empty frame" <<
std::endl;
}
```



```

        stream.flush();
        stream.Close();
    };

    Aws::Utils::Threading::Semaphore signaling(0 /*initialCount*/, 1 /*maxCount*/);
    auto OnResponseCallback = [&signaling](
        const TranscribeStreamingServiceClient * /*unused*/,
        const Model::StartStreamTranscriptionRequest & /*unused*/,
        const Model::StartStreamTranscriptionOutcome &outcome,
        const std::shared_ptr<const Aws::Client::AsyncCallerContext> & /*unused*/) {

        if (!outcome.IsSuccess()) {
            std::cerr << "Transcribe streaming error "
                << outcome.GetError().GetMessage() << std::endl;
        }

        signaling.Release();
    };

    std::cout << "Starting..." << std::endl;
    client.StartStreamTranscriptionAsync(request, OnStreamReady,
    OnResponseCallback,
                                     nullptr /*context*/);
    signaling.WaitOne(); // Prevent the application from exiting until we're
    done.
    std::cout << "Done" << std::endl;
}

Aws::ShutdownAPI(options);

return 0;
}

```

- Para obter detalhes da API, consulte [StartStreamTranscriptionAsync](#) Referência AWS SDK for C++ da API.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use `StartTranscriptionJob` com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `StartTranscriptionJob`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação em contexto nos seguintes exemplos de código:

- [Criar e refinar um vocabulário personalizado](#)
- [Transcrever áudio e obter dados do trabalho](#)

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// Start a transcription job for a media file. This method returns
/// as soon as the job is started.
/// </summary>
/// <param name="jobName">A unique name for the transcription job.</param>
/// <param name="mediaFileUri">The URI of the media file, typically an Amazon
S3 location.</param>
/// <param name="mediaFormat">The format of the media file.</param>
/// <param name="languageCode">The language code of the media file, such as
en-US.</param>
/// <param name="vocabularyName">Optional name of a custom vocabulary.</
param>
/// <returns>A TranscriptionJob instance with information on the new job.</
returns>
public async Task<TranscriptionJob> StartTranscriptionJob(string jobName,
string mediaFileUri,
    MediaFormat mediaFormat, LanguageCode languageCode, string?
vocabularyName)
{
    var response = await _amazonTranscribeService.StartTranscriptionJobAsync(
```

```
new StartTranscriptionJobRequest()
{
    TranscriptionJobName = jobName,
    Media = new Media()
    {
        MediaFileUri = mediaFileUri
    },
    MediaFormat = mediaFormat,
    LanguageCode = languageCode,
    Settings = vocabularyName != null ? new Settings()
    {
        VocabularyName = vocabularyName
    } : null
});
return response.TranscriptionJob;
}
```

- Para obter detalhes da API, consulte [StartTranscriptionJob](#) Referência AWS SDK for .NET da API.

CLI

AWS CLI

Exemplo 1: como transcrever um arquivo de áudio

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio.

```
aws transcribe start-transcription-job \
  --cli-input-json file://myfile.json
```

Conteúdo de `myfile.json`:

```
{
  "TranscriptionJobName": "cli-simple-transcription-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
  }
}
```

```
}
```

Para obter mais informações, consulte [Conceitos básicos \(interface de linha de AWS comando\)](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 2: como transcrever um arquivo de áudio multicanal

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio multicanal.

```
aws transcribe start-transcription-job \  
  --cli-input-json file://mysecondfile.json
```

Conteúdo de `mysecondfile.json`:

```
{  
  "TranscriptionJobName": "cli-channelid-job",  
  "LanguageCode": "the-language-of-your-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
  },  
  "Settings": {  
    "ChannelIdentification": true  
  }  
}
```

Saída:

```
{  
  "TranscriptionJob": {  
    "TranscriptionJobName": "cli-channelid-job",  
    "TranscriptionJobStatus": "IN_PROGRESS",  
    "LanguageCode": "the-language-of-your-transcription-job",  
    "Media": {  
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-  
file-name.file-extension"  
    },  
    "StartTime": "2020-09-17T16:07:56.817000+00:00",  
    "CreationTime": "2020-09-17T16:07:56.784000+00:00",  
    "Settings": {  
      "ChannelIdentification": true  
    }  
  }  
}
```

```

    }
  }
}

```

Para obter mais informações, consulte [Transcribing Multi-Channel Audio](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 3: como transcrever um arquivo de áudio e identificar quem está falando

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio e identifica os locutores na saída da transcrição.

```

aws transcribe start-transcription-job \
  --cli-input-json file://mythirdfile.json

```

Conteúdo de `mythirdfile.json`:

```

{
  "TranscriptionJobName": "cli-speakerid-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
  },
  "Settings":{
    "ShowSpeakerLabels": true,
    "MaxSpeakerLabels": 2
  }
}

```

Saída:

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-speakerid-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-17T16:22:59.696000+00:00",

```

```

    "CreationTime": "2020-09-17T16:22:59.676000+00:00",
    "Settings": {
      "ShowSpeakerLabels": true,
      "MaxSpeakerLabels": 2
    }
  }
}

```

Para obter mais informações, consulte [Identifying Speakers](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 4: como transcrever um arquivo de áudio e mascarar palavras indesejadas na saída da transcrição

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio e usa um filtro de vocabulário que você criou anteriormente para mascarar palavras indesejadas.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myfourthfile.json

```

Conteúdo de `myfourthfile.json`:

```

{
  "TranscriptionJobName": "cli-filter-mask-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-name.file-extension"
  },
  "Settings":{
    "VocabularyFilterName": "your-vocabulary-filter",
    "VocabularyFilterMethod": "mask"
  }
}

```

Saída:

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-filter-mask-job",
    "TranscriptionJobStatus": "IN_PROGRESS",

```

```

    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-
extension"
    },
    "StartTime": "2020-09-18T16:36:18.568000+00:00",
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
      "VocabularyFilterName": "your-vocabulary-filter",
      "VocabularyFilterMethod": "mask"
    }
  }
}

```

Para obter mais informações, consulte [Filtering Transcriptions](#) o Guia do desenvolvedor do Amazon Transcribe.

Exemplo 5: como transcrever um arquivo de áudio e remover palavras indesejadas na saída da transcrição

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio e usa um filtro de vocabulário que você criou anteriormente para mascarar palavras indesejadas.

```

aws transcribe start-transcription-job \
  --cli-input-json file://myfifthfile.json

```

Conteúdo de `myfifthfile.json`:

```

{
  "TranscriptionJobName": "cli-filter-remove-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
  },
  "Settings":{
    "VocabularyFilterName": "your-vocabulary-filter",
    "VocabularyFilterMethod": "remove"
  }
}

```

Saída:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-filter-remove-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-18T16:36:18.568000+00:00",
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
      "VocabularyFilterName": "your-vocabulary-filter",
      "VocabularyFilterMethod": "remove"
    }
  }
}
```

Para obter mais informações, consulte [Filtering Transcriptions](#) o Guia do desenvolvedor do Amazon Transcribe.

Exemplo 6: como transcrever um arquivo de áudio com maior precisão usando um vocabulário personalizado

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio e usa um filtro de vocabulário que você criou anteriormente para mascarar palavras indesejadas.

```
aws transcribe start-transcription-job \
  --cli-input-json file://mysixthfile.json
```

Conteúdo de `mysixthfile.json`:

```
{
  "TranscriptionJobName": "cli-vocab-job",
  "LanguageCode": "the-language-of-your-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
  },
  "Settings":{
    "VocabularyName": "your-vocabulary"
  }
}
```



```
}
```

Saída:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-vocab-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "StartTime": "2020-09-18T16:36:18.568000+00:00",
    "CreationTime": "2020-09-18T16:36:18.547000+00:00",
    "Settings": {
      "VocabularyName": "your-vocabulary"
    }
  }
}
```

Para obter mais informações, consulte [Filtering Transcriptions](#) o Guia do desenvolvedor do Amazon Transcribe.

Exemplo 7: como identificar o idioma de um arquivo de áudio e transcrevê-lo

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio e usa um filtro de vocabulário que você criou anteriormente para mascarar palavras indesejadas.

```
aws transcribe start-transcription-job \
  --cli-input-json file://myseventhfile.json
```

Conteúdo de `myseventhfile.json`:

```
{
  "TranscriptionJobName": "cli-identify-language-transcription-job",
  "IdentifyLanguage": true,
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-
file-name.file-extension"
  }
}
```

Saída:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-identify-language-transcription-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/Amazon-S3-prefix/your-media-file-name.file-extension"
    },
    "StartTime": "2020-09-18T22:27:23.970000+00:00",
    "CreationTime": "2020-09-18T22:27:23.948000+00:00",
    "IdentifyLanguage": true
  }
}
```

Para obter mais informações, consulte [Identifying the Language](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 8: como transcrever um arquivo de áudio com informações de identificação pessoal censuradas

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio e oculta qualquer informação de identificação pessoal na saída da transcrição.

```
aws transcribe start-transcription-job \
  --cli-input-json file://myeighthfile.json
```

Conteúdo de `myeighthfile.json`:

```
{
  "TranscriptionJobName": "cli-redaction-job",
  "LanguageCode": "language-code",
  "Media": {
    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
  },
  "ContentRedaction": {
    "RedactionOutput": "redacted",
    "RedactionType": "PII"
  }
}
```

Saída:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-redaction-job",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-
extension"
    },
    "StartTime": "2020-09-25T23:49:13.195000+00:00",
    "CreationTime": "2020-09-25T23:49:13.176000+00:00",
    "ContentRedaction": {
      "RedactionType": "PII",
      "RedactionOutput": "redacted"
    }
  }
}
```

Para obter mais informações, consulte [Automatic Content Redaction](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 9: como gerar uma transcrição com informações de identificação pessoal (PII) censuradas e uma transcrição não censurada

O exemplo de `start-transcription-job` a seguir gera duas transcrições do arquivo de áudio, uma com as informações de identificação pessoal editadas e a outra sem nenhuma edição.

```
aws transcribe start-transcription-job \
  --cli-input-json file://myninthfile.json
```

Conteúdo de `myninthfile.json`:

```
{
  "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",
  "LanguageCode": "language-code",
  "Media": {
    "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
  },
  "ContentRedaction": {
```

```

    "RedactionOutput": "redacted_and_unredacted",
    "RedactionType": "PII"
  }
}

```

Saída:

```

{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-
extension"
    },
    "StartTime": "2020-09-25T23:59:47.677000+00:00",
    "CreationTime": "2020-09-25T23:59:47.653000+00:00",
    "ContentRedaction": {
      "RedactionType": "PII",
      "RedactionOutput": "redacted_and_unredacted"
    }
  }
}

```

Para obter mais informações, consulte [Automatic Content Redaction](#) no Guia do desenvolvedor do Amazon Transcribe.

Exemplo 10: como usar um modelo de linguagem personalizado criado anteriormente para transcrever um arquivo de áudio.

O exemplo de `start-transcription-job` a seguir transcreve o arquivo de áudio com um modelo de idioma personalizado que você criou anteriormente.

```

aws transcribe start-transcription-job \
  --cli-input-json file://mytenthfile.json

```

Conteúdo de `mytenthfile.json`:

```

{
  "TranscriptionJobName": "cli-clm-2-job-1",
  "LanguageCode": "language-code",

```

```
"Media": {
  "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-extension"
},
"ModelSettings": {
  "LanguageModelName": "cli-clm-2"
}
}
```

Saída:

```
{
  "TranscriptionJob": {
    "TranscriptionJobName": "cli-clm-2-job-1",
    "TranscriptionJobStatus": "IN_PROGRESS",
    "LanguageCode": "language-code",
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/your-audio-file.file-
extension"
    },
    "StartTime": "2020-09-28T17:56:01.835000+00:00",
    "CreationTime": "2020-09-28T17:56:01.801000+00:00",
    "ModelSettings": {
      "LanguageModelName": "cli-clm-2"
    }
  }
}
```

Para obter mais informações, consulte [Improving Domain-Specific Transcription Accuracy with Custom Language Models](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [StartTranscriptionJob](#) em Referência de AWS CLI Comandos.

Java

SDK para Java 2.x

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
public class TranscribeStreamingDemoApp {
    private static final Region REGION = Region.US_EAST_1;
    private static TranscribeStreamingAsyncClient client;

    public static void main(String args[])
        throws URISyntaxException, ExecutionException, InterruptedException,
        LineUnavailableException {

        client = TranscribeStreamingAsyncClient.builder()
            .credentialsProvider(getCredentials())
            .region(REGION)
            .build();

        CompletableFuture<Void> result =
client.startStreamTranscription(getRequest(16_000),
    new AudioStreamPublisher(getStreamFromMic()),
    getResponseHandler());

        result.get();
        client.close();
    }

    private static InputStream getStreamFromMic() throws LineUnavailableException
    {

        // Signed PCM AudioFormat with 16kHz, 16 bit sample size, mono
        int sampleRate = 16000;
        AudioFormat format = new AudioFormat(sampleRate, 16, 1, true, false);
        DataLine.Info info = new DataLine.Info(TargetDataLine.class, format);

        if (!AudioSystem.isLineSupported(info)) {
            System.out.println("Line not supported");
            System.exit(0);
        }

        TargetDataLine line = (TargetDataLine) AudioSystem.getLine(info);
        line.open(format);
        line.start();

        InputStream audioStream = new AudioInputStream(line);
        return audioStream;
    }
}
```

```
private static AwsCredentialsProvider getCredentials() {
    return DefaultCredentialsProvider.create();
}

private static StartStreamTranscriptionRequest getRequest(Integer
mediaSampleRateHertz) {
    return StartStreamTranscriptionRequest.builder()
        .languageCode(LanguageCode.EN_US.toString())
        .mediaEncoding(MediaEncoding.PCM)
        .mediaSampleRateHertz(mediaSampleRateHertz)
        .build();
}

private static StartStreamTranscriptionResponseHandler getResponseHandler() {
    return StartStreamTranscriptionResponseHandler.builder()
        .onResponse(r -> {
            System.out.println("Received Initial response");
        })
        .onError(e -> {
            System.out.println(e.getMessage());
            StringWriter sw = new StringWriter();
            e.printStackTrace(new PrintWriter(sw));
            System.out.println("Error Occurred: " + sw.toString());
        })
        .onComplete(() -> {
            System.out.println("=== All records stream successfully
===");
        })
        .subscriber(event -> {
            List<Result> results = ((TranscriptEvent)
event).transcript().results();
            if (results.size() > 0) {
                if (!
results.get(0).alternatives().get(0).transcript().isEmpty()) {

System.out.println(results.get(0).alternatives().get(0).transcript());
                }
            }
        })
        .build();
}

private InputStream getStreamFromFile(String audioFileName) {
    try {
```

```
        File inputFile = new
File(getClass().getClassLoader().getResource(audioFileName).getFile());
        InputStream audioStream = new FileInputStream(inputFile);
        return audioStream;
    } catch (FileNotFoundException e) {
        throw new RuntimeException(e);
    }
}

private static class AudioStreamPublisher implements Publisher<AudioStream> {
    private final InputStream inputStream;
    private static Subscription currentSubscription;

    private AudioStreamPublisher(InputStream inputStream) {
        this.inputStream = inputStream;
    }

    @Override
    public void subscribe(Subscriber<? super AudioStream> s) {

        if (this.currentSubscription == null) {
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        } else {
            this.currentSubscription.cancel();
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        }
        s.onSubscribe(currentSubscription);
    }
}

public static class SubscriptionImpl implements Subscription {
    private static final int CHUNK_SIZE_IN_BYTES = 1024 * 1;
    private final Subscriber<? super AudioStream> subscriber;
    private final InputStream inputStream;
    private ExecutorService executor = Executors.newFixedThreadPool(1);
    private AtomicLong demand = new AtomicLong(0);

    SubscriptionImpl(Subscriber<? super AudioStream> s, InputStream
inputStream) {
        this.subscriber = s;
        this.inputStream = inputStream;
    }

    @Override
```



```
public void request(long n) {
    if (n <= 0) {
        subscriber.onError(new IllegalArgumentException("Demand must be
positive"));
    }

    demand.getAndAdd(n);

    executor.submit(() -> {
        try {
            do {
                ByteBuffer audioBuffer = getNextEvent();
                if (audioBuffer.remaining() > 0) {
                    AudioEvent audioEvent =
audioEventFromBuffer(audioBuffer);
                    subscriber.onNext(audioEvent);
                } else {
                    subscriber.onComplete();
                    break;
                }
            } while (demand.decrementAndGet() > 0);
        } catch (Exception e) {
            subscriber.onError(e);
        }
    });
}

@Override
public void cancel() {
    executor.shutdown();
}

private ByteBuffer getNextEvent() {
    ByteBuffer audioBuffer = null;
    byte[] audioBytes = new byte[CHUNK_SIZE_IN_BYTES];

    int len = 0;
    try {
        len = inputStream.read(audioBytes);

        if (len <= 0) {
            audioBuffer = ByteBuffer.allocate(0);
        } else {
            audioBuffer = ByteBuffer.wrap(audioBytes, 0, len);
        }
    }
}
```

```
    }
    } catch (IOException e) {
        throw new UncheckedIOException(e);
    }

    return audioBuffer;
}

private AudioEvent audioEventFromBuffer(ByteBuffer bb) {
    return AudioEvent.builder()
        .audioChunk(SdkBytes.fromByteBuffer(bb))
        .build();
}
}
```

- Para obter detalhes da API, consulte [StartTranscriptionJob](#) Referência AWS SDK for Java 2.x da API.

JavaScript

SDK para JavaScript (v3)

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

Iniciar um trabalho de transcrição.

```
// Import the required AWS SDK clients and commands for Node.js
import { StartTranscriptionJobCommand } from "@aws-sdk/client-transcribe";
import { transcribeClient } from "../libs/transcribeClient.js";

// Set the parameters
export const params = {
    TranscriptionJobName: "JOB_NAME",
    LanguageCode: "LANGUAGE_CODE", // For example, 'en-US'
    MediaFormat: "SOURCE_FILE_FORMAT", // For example, 'wav'
```

```
Media: {
  MediaFileUri: "SOURCE_LOCATION",
  // For example, "https://transcribe-demo.s3-REGION.amazonaws.com/
hello_world.wav"
},
OutputBucketName: "OUTPUT_BUCKET_NAME"
};

export const run = async () => {
  try {
    const data = await transcribeClient.send(
      new StartTranscriptionJobCommand(params)
    );
    console.log("Success - put", data);
    return data; // For unit tests.
  } catch (err) {
    console.log("Error", err);
  }
};
run();
```

Crie o cliente.

```
import { TranscribeClient } from "@aws-sdk/client-transcribe";
// Set the AWS Region.
const REGION = "REGION"; //e.g. "us-east-1"
// Create an Amazon Transcribe service client object.
const transcribeClient = new TranscribeClient({ region: REGION });
export { transcribeClient };
```

- Para obter mais informações, consulte o [Guia do desenvolvedor do AWS SDK for JavaScript](#).
- Para obter detalhes da API, consulte [StartTranscriptionJob](#) Referência AWS SDK for JavaScript da API.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
def start_job(
    job_name,
    media_uri,
    media_format,
    language_code,
    transcribe_client,
    vocabulary_name=None,
):
    """
    Starts a transcription job. This function returns as soon as the job is
    started.

    To get the current status of the job, call get_transcription_job. The job is
    successfully completed when the job status is 'COMPLETED'.

    :param job_name: The name of the transcription job. This must be unique for
        your AWS account.
    :param media_uri: The URI where the audio file is stored. This is typically
        in an Amazon S3 bucket.
    :param media_format: The format of the audio file. For example, mp3 or wav.
    :param language_code: The language code of the audio file.
        For example, en-US or ja-JP
    :param transcribe_client: The Boto3 Transcribe client.
    :param vocabulary_name: The name of a custom vocabulary to use when
    transcribing
        the audio file.
    :return: Data about the job.
    """
    try:
        job_args = {
            "TranscriptionJobName": job_name,
            "Media": {"MediaFileUri": media_uri},
            "MediaFormat": media_format,
```

```
        "LanguageCode": language_code,
    }
    if vocabulary_name is not None:
        job_args["Settings"] = {"VocabularyName": vocabulary_name}
    response = transcribe_client.start_transcription_job(**job_args)
    job = response["TranscriptionJob"]
    logger.info("Started transcription job %s.", job_name)
except ClientError:
    logger.exception("Couldn't start transcription job %s.", job_name)
    raise
else:
    return job
```

- Para obter detalhes da API, consulte a [StartTranscriptionJob](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **UpdateVocabulary** com um AWS SDK ou CLI

Os exemplos de códigos a seguir mostram como usar `UpdateVocabulary`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação no contexto no seguinte exemplo de código:

- [Criar e refinar um vocabulário personalizado](#)

.NET

AWS SDK for .NET

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
/// <summary>
/// Update a custom vocabulary with new values. Update overwrites all
existing information.
/// </summary>
/// <param name="languageCode">The language code of the vocabulary.</param>
/// <param name="phrases">Phrases to use in the vocabulary.</param>
/// <param name="vocabularyName">Name for the vocabulary.</param>
/// <returns>The state of the custom vocabulary.</returns>
public async Task<VocabularyState> UpdateCustomVocabulary(LanguageCode
languageCode,
    List<string> phrases, string vocabularyName)
{
    var response = await _amazonTranscribeService.UpdateVocabularyAsync(
        new UpdateVocabularyRequest()
        {
            LanguageCode = languageCode,
            Phrases = phrases,
            VocabularyName = vocabularyName
        });
    return response.VocabularyState;
}
```

- Para obter detalhes da API, consulte [UpdateVocabulary](#) a Referência AWS SDK for .NET da API.

CLI

AWS CLI

Como atualizar um vocabulário personalizado com novos termos.

O exemplo de `update-vocabulary` a seguir substitui os termos usados para criar o vocabulário personalizado pelos novos termos fornecidos. Pré-requisito: para substituir os termos em um vocabulário personalizado, você precisa de um arquivo com novos termos.

```
aws transcribe update-vocabulary \
    --vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/Amazon-S3-Prefix/custom-
vocabulary.txt \
    --vocabulary-name custom-vocabulary \
```

```
--language-code language-code
```

Saída:

```
{
  "VocabularyName": "custom-vocabulary",
  "LanguageCode": "language",
  "VocabularyState": "PENDING"
}
```

Para obter mais informações, consulte [Custom Vocabularies](#) no Guia do desenvolvedor do Amazon Transcribe.

- Para obter detalhes da API, consulte [UpdateVocabulary](#) em Referência de AWS CLI Comandos.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
def update_vocabulary(
    vocabulary_name, language_code, transcribe_client, phrases=None,
    table_uri=None
):
    """
    Updates an existing custom vocabulary. The entire vocabulary is replaced with
    the contents of the update.

    :param vocabulary_name: The name of the vocabulary to update.
    :param language_code: The language code of the vocabulary.
    :param transcribe_client: The Boto3 Transcribe client.
    :param phrases: A list of comma-separated phrases to include in the
    vocabulary.
    :param table_uri: A table of phrases and pronunciation hints to include in
    the
```

```
        vocabulary.  
        """  
        try:  
            vocab_args = {"VocabularyName": vocabulary_name, "LanguageCode":  
language_code}  
            if phrases is not None:  
                vocab_args["Phrases"] = phrases  
            elif table_uri is not None:  
                vocab_args["VocabularyFileUri"] = table_uri  
            response = transcribe_client.update_vocabulary(**vocab_args)  
            logger.info("Updated custom vocabulary %s.", response["VocabularyName"])  
        except ClientError:  
            logger.exception("Couldn't update custom vocabulary %s.",  
vocabulary_name)  
            raise
```

- Para obter detalhes da API, consulte a [UpdateVocabulary](#) Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Cenários para o Amazon Transcribe usando SDKs AWS

Os exemplos de código a seguir mostram como implementar cenários comuns no Amazon AWS Transcribe com SDKs. Esses cenários mostram como realizar tarefas específicas chamando várias funções no Amazon Transcribe. Cada cenário inclui um link para GitHub, onde você pode encontrar instruções sobre como configurar e executar o código.

Exemplos

- [Crie e refine um vocabulário personalizado do Amazon Transcribe usando um SDK AWS](#)
- [Transcreva áudio e obtenha dados de trabalho com o Amazon Transcribe usando um SDK AWS](#)

Crie e refine um vocabulário personalizado do Amazon Transcribe usando um SDK AWS

O exemplo de código a seguir mostra como:

- Fazer upload de um arquivo de áudio para o Amazon S3.
- Executar um trabalho do Amazon Transcribe para transcrever o arquivo e obter os resultados.
- Criar e refinar um vocabulário personalizado para melhorar a precisão da transcrição.
- Executar trabalhos com vocabulários personalizados e obter os resultados.

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

Transcreva um arquivo de áudio que contenha uma leitura de Jabberwocky de Lewis Carroll. Comece criando funções que envolvam as ações do Amazon Transcribe.

```
def start_job(
    job_name,
    media_uri,
    media_format,
    language_code,
    transcribe_client,
    vocabulary_name=None,
):
    """
    Starts a transcription job. This function returns as soon as the job is
    started.

    To get the current status of the job, call get_transcription_job. The job is
    successfully completed when the job status is 'COMPLETED'.

    :param job_name: The name of the transcription job. This must be unique for
        your AWS account.
    :param media_uri: The URI where the audio file is stored. This is typically
```

```
        in an Amazon S3 bucket.
:param media_format: The format of the audio file. For example, mp3 or wav.
:param language_code: The language code of the audio file.
                    For example, en-US or ja-JP
:param transcribe_client: The Boto3 Transcribe client.
:param vocabulary_name: The name of a custom vocabulary to use when
transcribing
                    the audio file.
:return: Data about the job.
"""
try:
    job_args = {
        "TranscriptionJobName": job_name,
        "Media": {"MediaFileUri": media_uri},
        "MediaFormat": media_format,
        "LanguageCode": language_code,
    }
    if vocabulary_name is not None:
        job_args["Settings"] = {"VocabularyName": vocabulary_name}
    response = transcribe_client.start_transcription_job(**job_args)
    job = response["TranscriptionJob"]
    logger.info("Started transcription job %s.", job_name)
except ClientError:
    logger.exception("Couldn't start transcription job %s.", job_name)
    raise
else:
    return job

def get_job(job_name, transcribe_client):
    """
    Gets details about a transcription job.

    :param job_name: The name of the job to retrieve.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: The retrieved transcription job.
    """
    try:
        response = transcribe_client.get_transcription_job(
            TranscriptionJobName=job_name
        )
        job = response["TranscriptionJob"]
        logger.info("Got job %s.", job["TranscriptionJobName"])
```

```
except ClientError:
    logger.exception("Couldn't get job %s.", job_name)
    raise
else:
    return job

def delete_job(job_name, transcribe_client):
    """
    Deletes a transcription job. This also deletes the transcript associated with
    the job.

    :param job_name: The name of the job to delete.
    :param transcribe_client: The Boto3 Transcribe client.
    """
    try:
        transcribe_client.delete_transcription_job(TranscriptionJobName=job_name)
        logger.info("Deleted job %s.", job_name)
    except ClientError:
        logger.exception("Couldn't delete job %s.", job_name)
        raise

def create_vocabulary(
    vocabulary_name, language_code, transcribe_client, phrases=None,
    table_uri=None
):
    """
    Creates a custom vocabulary that can be used to improve the accuracy of
    transcription jobs. This function returns as soon as the vocabulary
    processing
    is started. Call get_vocabulary to get the current status of the vocabulary.
    The vocabulary is ready to use when its status is 'READY'.

    :param vocabulary_name: The name of the custom vocabulary.
    :param language_code: The language code of the vocabulary.
        For example, en-US or nl-NL.
    :param transcribe_client: The Boto3 Transcribe client.
    :param phrases: A list of comma-separated phrases to include in the
    vocabulary.
    :param table_uri: A table of phrases and pronunciation hints to include in
    the
```

```
        vocabulary.  
:return: Information about the newly created vocabulary.  
""  
try:  
    vocab_args = {"VocabularyName": vocabulary_name, "LanguageCode":  
language_code}  
    if phrases is not None:  
        vocab_args["Phrases"] = phrases  
    elif table_uri is not None:  
        vocab_args["VocabularyFileUri"] = table_uri  
    response = transcribe_client.create_vocabulary(**vocab_args)  
    logger.info("Created custom vocabulary %s.", response["VocabularyName"])  
except ClientError:  
    logger.exception("Couldn't create custom vocabulary %s.",  
vocabulary_name)  
    raise  
else:  
    return response  
  
def get_vocabulary(vocabulary_name, transcribe_client):  
    ""  
    Gets information about a custom vocabulary.  
  
    :param vocabulary_name: The name of the vocabulary to retrieve.  
    :param transcribe_client: The Boto3 Transcribe client.  
    :return: Information about the vocabulary.  
    ""  
    try:  
        response =  
transcribe_client.get_vocabulary(VocabularyName=vocabulary_name)  
        logger.info("Got vocabulary %s.", response["VocabularyName"])  
    except ClientError:  
        logger.exception("Couldn't get vocabulary %s.", vocabulary_name)  
        raise  
    else:  
        return response  
  
def update_vocabulary(  
    vocabulary_name, language_code, transcribe_client, phrases=None,  
    table_uri=None
```

```

):
    """
    Updates an existing custom vocabulary. The entire vocabulary is replaced with
    the contents of the update.

    :param vocabulary_name: The name of the vocabulary to update.
    :param language_code: The language code of the vocabulary.
    :param transcribe_client: The Boto3 Transcribe client.
    :param phrases: A list of comma-separated phrases to include in the
vocabulary.
    :param table_uri: A table of phrases and pronunciation hints to include in
the
                    vocabulary.
    """
    try:
        vocab_args = {"VocabularyName": vocabulary_name, "LanguageCode":
language_code}
        if phrases is not None:
            vocab_args["Phrases"] = phrases
        elif table_uri is not None:
            vocab_args["VocabularyFileUri"] = table_uri
        response = transcribe_client.update_vocabulary(**vocab_args)
        logger.info("Updated custom vocabulary %s.", response["VocabularyName"])
    except ClientError:
        logger.exception("Couldn't update custom vocabulary %s.",
vocabulary_name)
        raise

def list_vocabularies(vocabulary_filter, transcribe_client):
    """
    Lists the custom vocabularies created for this AWS account.

    :param vocabulary_filter: The returned vocabularies must contain this string
in
                    their names.
    :param transcribe_client: The Boto3 Transcribe client.
    :return: The list of retrieved vocabularies.
    """
    try:
        response =
transcribe_client.list_vocabularies(NameContains=vocabulary_filter)
        vocabs = response["Vocabularies"]

```

```

next_token = response.get("NextToken")
while next_token is not None:
    response = transcribe_client.list_vocabularies(
        NameContains=vocabulary_filter, NextToken=next_token
    )
    vocabs += response["Vocabularies"]
    next_token = response.get("NextToken")
logger.info(
    "Got %s vocabularies with filter %s.", len(vocabs), vocabulary_filter
)
except ClientError:
    logger.exception(
        "Couldn't list vocabularies with filter %s.", vocabulary_filter
    )
    raise
else:
    return vocabs

```

```

def delete_vocabulary(vocabulary_name, transcribe_client):
    """
    Deletes a custom vocabulary.

    :param vocabulary_name: The name of the vocabulary to delete.
    :param transcribe_client: The Boto3 Transcribe client.
    """
    try:
        transcribe_client.delete_vocabulary(VocabularyName=vocabulary_name)
        logger.info("Deleted vocabulary %s.", vocabulary_name)
    except ClientError:
        logger.exception("Couldn't delete vocabulary %s.", vocabulary_name)
        raise

```

Chame as funções do wrapper para transcrever áudio sem um vocabulário personalizado e, em seguida, com diferentes versões de um vocabulário personalizado para ver os melhores resultados.

```

def usage_demo():
    """Shows how to use the Amazon Transcribe service."""

```

```
logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

s3_resource = boto3.resource("s3")
transcribe_client = boto3.client("transcribe")

print("-" * 88)
print("Welcome to the Amazon Transcribe demo!")
print("-" * 88)

bucket_name = f"jabber-bucket-{time.time_ns()}"
print(f"Creating bucket {bucket_name}.")
bucket = s3_resource.create_bucket(
    Bucket=bucket_name,
    CreateBucketConfiguration={
        "LocationConstraint": transcribe_client.meta.region_name
    },
)
media_file_name = ".media/Jabberwocky.mp3"
media_object_key = "Jabberwocky.mp3"
print(f"Uploading media file {media_file_name}.")
bucket.upload_file(media_file_name, media_object_key)
media_uri = f"s3://{bucket.name}/{media_object_key}"

job_name_simple = f"Jabber-{time.time_ns()}"
print(f"Starting transcription job {job_name_simple}.")
start_job(
    job_name_simple,
    f"s3://{bucket.name}/{media_object_key}",
    "mp3",
    "en-US",
    transcribe_client,
)
transcribe_waiter = TranscribeCompleteWaiter(transcribe_client)
transcribe_waiter.wait(job_name_simple)
job_simple = get_job(job_name_simple, transcribe_client)
transcript_simple = requests.get(
    job_simple["Transcript"]["TranscriptFileUri"]
).json()
print(f"Transcript for job {transcript_simple['jobName']}:")
print(transcript_simple["results"]["transcripts"][0]["transcript"])

print("-" * 88)
print(
    "Creating a custom vocabulary that lists the nonsense words to try to "
```

```
        "improve the transcription."
    )
    vocabulary_name = f"Jabber-vocabulary-{time.time_ns()}"
    create_vocabulary(
        vocabulary_name,
        "en-US",
        transcribe_client,
        phrases=[
            "brillig",
            "slithy",
            "borogoves",
            "mome",
            "raths",
            "Jub-Jub",
            "frumious",
            "manxome",
            "Tumtum",
            "uffish",
            "whiffling",
            "tulgey",
            "thou",
            "frabjous",
            "callooh",
            "callay",
            "chortled",
        ],
    )
    vocabulary_ready_waiter = VocabularyReadyWaiter(transcribe_client)
    vocabulary_ready_waiter.wait(vocabulary_name)

    job_name_vocabulary_list = f"Jabber-vocabulary-list-{time.time_ns()}"
    print(f"Starting transcription job {job_name_vocabulary_list}.")
    start_job(
        job_name_vocabulary_list,
        media_uri,
        "mp3",
        "en-US",
        transcribe_client,
        vocabulary_name,
    )
    transcribe_waiter.wait(job_name_vocabulary_list)
    job_vocabulary_list = get_job(job_name_vocabulary_list, transcribe_client)
    transcript_vocabulary_list = requests.get(
        job_vocabulary_list["Transcript"]["TranscriptFileUri"]
```



```

).json()
print(f"Transcript for job {transcript_vocabulary_list['jobName']}:")
print(transcript_vocabulary_list["results"]["transcripts"][0]["transcript"])

print("-" * 88)
print(
    "Updating the custom vocabulary with table data that provides additional
"
    "pronunciation hints."
)
table_vocab_file = "jabber-vocabulary-table.txt"
bucket.upload_file(table_vocab_file, table_vocab_file)
update_vocabulary(
    vocabulary_name,
    "en-US",
    transcribe_client,
    table_uri=f"s3://{bucket.name}/{table_vocab_file}",
)
vocabulary_ready_waiter.wait(vocabulary_name)

job_name_vocab_table = f"Jabber-vocab-table-{time.time_ns()}"
print(f"Starting transcription job {job_name_vocab_table}.")
start_job(
    job_name_vocab_table,
    media_uri,
    "mp3",
    "en-US",
    transcribe_client,
    vocabulary_name=vocabulary_name,
)
transcribe_waiter.wait(job_name_vocab_table)
job_vocab_table = get_job(job_name_vocab_table, transcribe_client)
transcript_vocab_table = requests.get(
    job_vocab_table["Transcript"]["TranscriptFileUri"]
).json()
print(f"Transcript for job {transcript_vocab_table['jobName']}:")
print(transcript_vocab_table["results"]["transcripts"][0]["transcript"])

print("-" * 88)
print("Getting data for jobs and vocabularies.")
jabber_jobs = list_jobs("Jabber", transcribe_client)
print(f"Found {len(jabber_jobs)} jobs:")
for job_sum in jabber_jobs:
    job = get_job(job_sum["TranscriptionJobName"], transcribe_client)

```

```
print(
    f"\t{job['TranscriptionJobName']}, {job['Media']['MediaFileUri']}, "
    f"{job['Settings'].get('VocabularyName')}"
)

jabber_vocabs = list_vocabularies("Jabber", transcribe_client)
print(f"Found {len(jabber_vocabs)} vocabularies:")
for vocab_sum in jabber_vocabs:
    vocab = get_vocabulary(vocab_sum["VocabularyName"], transcribe_client)
    vocab_content = requests.get(vocab["DownloadUri"]).text
    print(f"\t{vocab['VocabularyName']} contents:")
    print(vocab_content)

print("-" * 88)
print("Deleting demo jobs.")
for job_name in [job_name_simple, job_name_vocabulary_list,
job_name_vocab_table]:
    delete_job(job_name, transcribe_client)
    print("Deleting demo vocabulary.")
    delete_vocabulary(vocabulary_name, transcribe_client)
    print("Deleting demo bucket.")
    bucket.objects.delete()
    bucket.delete()
    print("Thanks for watching!")
```

- Para obter detalhes da API, consulte os tópicos a seguir na Referência da API AWS SDK para Python (Boto3).
 - [CreateVocabulary](#)
 - [DeleteTranscriptionJob](#)
 - [DeleteVocabulary](#)
 - [GetTranscriptionJob](#)
 - [GetVocabulary](#)
 - [ListVocabularies](#)
 - [StartTranscriptionJob](#)
 - [UpdateVocabulary](#)

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Transcreva áudio e obtenha dados de trabalho com o Amazon Transcribe usando um SDK AWS

Os exemplos de código a seguir mostram como:

- Iniciar um trabalho de transcrição com o Amazon Transcribe.
- Aguardar a conclusão do trabalho.
- Obter o URI em que a transcrição está armazenada.

Para obter mais informações, consulte [Conceitos básicos do Amazon Transcribe](#).

Java

SDK para Java 2.x

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

Transcreve um arquivo PCM.

```
/**
 * To run this AWS code example, ensure that you have set up your development
 * environment, including your AWS credentials.
 *
 * For information, see this documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
 * started.html
 */

public class TranscribeStreamingDemoFile {
    private static final Region REGION = Region.US_EAST_1;
    private static TranscribeStreamingAsyncClient client;
```

```
public static void main(String args[]) throws ExecutionException,
InterruptedException {

    final String USAGE = "\n" +
        "Usage:\n" +
        "  <file> \n\n" +
        "Where:\n" +
        "  file - the location of a PCM file to transcribe. In this
example, ensure the PCM file is 16 hertz (Hz). \n";

    if (args.length != 1) {
        System.out.println(USAGE);
        System.exit(1);
    }

    String file = args[0];
    client = TranscribeStreamingAsyncClient.builder()
        .region(REGION)
        .build();

    CompletableFuture<Void> result =
client.startStreamTranscription(getRequest(16_000),
    new AudioStreamPublisher(getStreamFromFile(file)),
    getResponseHandler());

    result.get();
    client.close();
}

private static InputStream getStreamFromFile(String file) {
    try {
        File inputFile = new File(file);
        InputStream audioStream = new FileInputStream(inputFile);
        return audioStream;

    } catch (FileNotFoundException e) {
        throw new RuntimeException(e);
    }
}

private static StartStreamTranscriptionRequest getRequest(Integer
mediaSampleRateHertz) {
    return StartStreamTranscriptionRequest.builder()
```

```

        .languageCode(LanguageCode.EN_US)
        .mediaEncoding(MediaEncoding.PCM)
        .mediaSampleRateHertz(mediaSampleRateHertz)
        .build();
    }

    private static StartStreamTranscriptionResponseHandler getResponseHandler() {
        return StartStreamTranscriptionResponseHandler.builder()
            .onResponse(r -> {
                System.out.println("Received Initial response");
            })
            .onError(e -> {
                System.out.println(e.getMessage());
                StringWriter sw = new StringWriter();
                e.printStackTrace(new PrintWriter(sw));
                System.out.println("Error Occurred: " + sw.toString());
            })
            .onComplete(() -> {
                System.out.println("=== All records stream successfully
===");
            })
            .subscriber(event -> {
                List<Result> results = ((TranscriptEvent)
event).transcript().results();
                if (results.size() > 0) {
                    if (!
results.get(0).alternatives().get(0).transcript().isEmpty()) {

                        System.out.println(results.get(0).alternatives().get(0).transcript());
                    }
                }
            })
            .build();
    }

    private static class AudioStreamPublisher implements Publisher<AudioStream> {
        private final InputStream inputStream;
        private static Subscription currentSubscription;

        private AudioStreamPublisher(InputStream inputStream) {
            this.inputStream = inputStream;
        }

        @Override

```

```
public void subscribe(Subscriber<? super AudioStream> s) {

    if (this.currentSubscription == null) {
        this.currentSubscription = new SubscriptionImpl(s, inputStream);
    } else {
        this.currentSubscription.cancel();
        this.currentSubscription = new SubscriptionImpl(s, inputStream);
    }
    s.onSubscribe(currentSubscription);
}

}

public static class SubscriptionImpl implements Subscription {
    private static final int CHUNK_SIZE_IN_BYTES = 1024 * 1;
    private final Subscriber<? super AudioStream> subscriber;
    private final InputStream inputStream;
    private ExecutorService executor = Executors.newFixedThreadPool(1);
    private AtomicLong demand = new AtomicLong(0);

    SubscriptionImpl(Subscriber<? super AudioStream> s, InputStream
inputStream) {
        this.subscriber = s;
        this.inputStream = inputStream;
    }

    @Override
    public void request(long n) {
        if (n <= 0) {
            subscriber.onError(new IllegalArgumentException("Demand must be
positive"));
        }

        demand.getAndAdd(n);

        executor.submit(() -> {
            try {
                do {
                    ByteBuffer audioBuffer = getNextEvent();
                    if (audioBuffer.remaining() > 0) {
                        AudioEvent audioEvent =
audioEventFromBuffer(audioBuffer);
                        subscriber.onNext(audioEvent);
                    } else {
                        subscriber.onComplete();
                    }
                } while (demand.get() > 0);
            } catch (Exception e) {
                subscriber.onError(e);
            }
        });
    }
}
```

```
                break;
            }
        } while (demand.decrementAndGet() > 0);
    } catch (Exception e) {
        subscriber.onError(e);
    }
});
}

@Override
public void cancel() {
    executor.shutdown();
}

private ByteBuffer getNextEvent() {
    ByteBuffer audioBuffer = null;
    byte[] audioBytes = new byte[CHUNK_SIZE_IN_BYTES];

    int len = 0;
    try {
        len = inputStream.read(audioBytes);

        if (len <= 0) {
            audioBuffer = ByteBuffer.allocate(0);
        } else {
            audioBuffer = ByteBuffer.wrap(audioBytes, 0, len);
        }
    } catch (IOException e) {
        throw new UncheckedIOException(e);
    }

    return audioBuffer;
}

private AudioEvent audioEventFromBuffer(ByteBuffer bb) {
    return AudioEvent.builder()
        .audioChunk(SdkBytes.fromByteBuffer(bb))
        .build();
}
}
```

Transcreve o streaming de áudio do microfone do computador.

```
public class TranscribeStreamingDemoApp {
    private static final Region REGION = Region.US_EAST_1;
    private static TranscribeStreamingAsyncClient client;

    public static void main(String args[])
        throws URISyntaxException, ExecutionException, InterruptedException,
        LineUnavailableException {

        client = TranscribeStreamingAsyncClient.builder()
            .credentialsProvider(getCredentials())
            .region(REGION)
            .build();

        CompletableFuture<Void> result =
        client.startStreamTranscription(getRequest(16_000),
            new AudioStreamPublisher(getStreamFromMic()),
            getResponseHandler());

        result.get();
        client.close();
    }

    private static InputStream getStreamFromMic() throws LineUnavailableException
    {

        // Signed PCM AudioFormat with 16kHz, 16 bit sample size, mono
        int sampleRate = 16000;
        AudioFormat format = new AudioFormat(sampleRate, 16, 1, true, false);
        DataLine.Info info = new DataLine.Info(TargetDataLine.class, format);

        if (!AudioSystem.isLineSupported(info)) {
            System.out.println("Line not supported");
            System.exit(0);
        }

        TargetDataLine line = (TargetDataLine) AudioSystem.getLine(info);
        line.open(format);
        line.start();

        InputStream audioStream = new AudioInputStream(line);
        return audioStream;
    }
}
```



```
private static AwsCredentialsProvider getCredentials() {
    return DefaultCredentialsProvider.create();
}

private static StartStreamTranscriptionRequest getRequest(Integer
mediaSampleRateHertz) {
    return StartStreamTranscriptionRequest.builder()
        .languageCode(LanguageCode.EN_US.toString())
        .mediaEncoding(MediaEncoding.PCM)
        .mediaSampleRateHertz(mediaSampleRateHertz)
        .build();
}

private static StartStreamTranscriptionResponseHandler getResponseHandler() {
    return StartStreamTranscriptionResponseHandler.builder()
        .onResponse(r -> {
            System.out.println("Received Initial response");
        })
        .onError(e -> {
            System.out.println(e.getMessage());
            StringWriter sw = new StringWriter();
            e.printStackTrace(new PrintWriter(sw));
            System.out.println("Error Occurred: " + sw.toString());
        })
        .onComplete(() -> {
            System.out.println("=== All records stream successfully
===");
        })
        .subscriber(event -> {
            List<Result> results = ((TranscriptEvent)
event).transcript().results();
            if (results.size() > 0) {
                if (!
results.get(0).alternatives().get(0).transcript().isEmpty()) {

System.out.println(results.get(0).alternatives().get(0).transcript());
                }
            }
        })
        .build();
}

private InputStream getStreamFromFile(String audioFileName) {
```

```
    try {
        File inputFile = new
File(getClass().getClassLoader().getResource(audioFileName).getFile());
        InputStream audioStream = new FileInputStream(inputFile);
        return audioStream;
    } catch (FileNotFoundException e) {
        throw new RuntimeException(e);
    }
}

private static class AudioStreamPublisher implements Publisher<AudioStream> {
    private final InputStream inputStream;
    private static Subscription currentSubscription;

    private AudioStreamPublisher(InputStream inputStream) {
        this.inputStream = inputStream;
    }

    @Override
    public void subscribe(Subscriber<? super AudioStream> s) {

        if (this.currentSubscription == null) {
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        } else {
            this.currentSubscription.cancel();
            this.currentSubscription = new SubscriptionImpl(s, inputStream);
        }
        s.onSubscribe(currentSubscription);
    }
}

public static class SubscriptionImpl implements Subscription {
    private static final int CHUNK_SIZE_IN_BYTES = 1024 * 1;
    private final Subscriber<? super AudioStream> subscriber;
    private final InputStream inputStream;
    private ExecutorService executor = Executors.newFixedThreadPool(1);
    private AtomicLong demand = new AtomicLong(0);

    SubscriptionImpl(Subscriber<? super AudioStream> s, InputStream
inputStream) {
        this.subscriber = s;
        this.inputStream = inputStream;
    }
}
```

```
@Override
public void request(long n) {
    if (n <= 0) {
        subscriber.onError(new IllegalArgumentException("Demand must be
positive"));
    }

    demand.getAndAdd(n);

    executor.submit(() -> {
        try {
            do {
                ByteBuffer audioBuffer = getNextEvent();
                if (audioBuffer.remaining() > 0) {
                    AudioEvent audioEvent =
audioEventFromBuffer(audioBuffer);
                    subscriber.onNext(audioEvent);
                } else {
                    subscriber.onComplete();
                    break;
                }
            } while (demand.decrementAndGet() > 0);
        } catch (Exception e) {
            subscriber.onError(e);
        }
    });
}

@Override
public void cancel() {
    executor.shutdown();
}

private ByteBuffer getNextEvent() {
    ByteBuffer audioBuffer = null;
    byte[] audioBytes = new byte[CHUNK_SIZE_IN_BYTES];

    int len = 0;
    try {
        len = inputStream.read(audioBytes);

        if (len <= 0) {
            audioBuffer = ByteBuffer.allocate(0);
        } else {
```

```
        audioBuffer = ByteBuffer.wrap(audioBytes, 0, len);
    }
} catch (IOException e) {
    throw new UncheckedIOException(e);
}

return audioBuffer;
}

private AudioEvent audioEventFromBuffer(ByteBuffer bb) {
    return AudioEvent.builder()
        .audioChunk(SdkBytes.fromByteBuffer(bb))
        .build();
}
}
}
```

- Para obter detalhes da API, consulte os tópicos a seguir na Referência da API AWS SDK for Java 2.x .
 - [GetTranscriptionJob](#)
 - [StartTranscriptionJob](#)

Python

SDK para Python (Boto3).

Note

Tem mais sobre GitHub. Encontre o exemplo completo e veja como configurar e executar no [AWS Code Examples Repository](#).

```
import time
import boto3

def transcribe_file(job_name, file_uri, transcribe_client):
    transcribe_client.start_transcription_job(
        TranscriptionJobName=job_name,
```

```
        Media={"MediaFileUri": file_uri},
        MediaFormat="wav",
        LanguageCode="en-US",
    )

    max_tries = 60
    while max_tries > 0:
        max_tries -= 1
        job =
transcribe_client.get_transcription_job(TranscriptionJobName=job_name)
        job_status = job["TranscriptionJob"]["TranscriptionJobStatus"]
        if job_status in ["COMPLETED", "FAILED"]:
            print(f"Job {job_name} is {job_status}.")
            if job_status == "COMPLETED":
                print(
                    f"Download the transcript from\n"
                    f"\t{job['TranscriptionJob']['Transcript']
['TranscriptFileUri']}."
                )
                break
            else:
                print(f"Waiting for {job_name}. Current status is {job_status}.")
                time.sleep(10)

def main():
    transcribe_client = boto3.client("transcribe")
    file_uri = "s3://test-transcribe/answer2.wav"
    transcribe_file("Example-job", file_uri, transcribe_client)

if __name__ == "__main__":
    main()
```

- Para obter detalhes da API, consulte os tópicos a seguir na Referência da API AWS SDK para Python (Boto3).
 - [GetTranscriptionJob](#)
 - [StartTranscriptionJob](#)

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Exemplos de serviços cruzados para o Amazon Transcribe usando SDKs AWS

Os exemplos de aplicativos a seguir usam AWS SDKs para combinar o Amazon Serviços da AWS Transcribe com outros. Cada exemplo inclui um link para GitHub, onde você pode encontrar instruções sobre como configurar e executar o aplicativo.

Exemplos

- [Criar uma aplicação Amazon Transcribe](#)
- [Criar uma aplicação de transmissão do Amazon Transcribe](#)
- [Converta texto em fala e vice-versa em texto usando um AWS SDK](#)

Criar uma aplicação Amazon Transcribe

O exemplo de código a seguir mostra como usar o Amazon Transcribe para transcrever e exibir gravações de voz no navegador.

JavaScript

SDK para JavaScript (v3)

Crie uma aplicação que use o Amazon Transcribe para transcrever e exibir gravações de voz no navegador. A aplicação usa dois buckets do Amazon Simple Storage Service (Amazon S3), um para hospedar o código da aplicação e outro para armazenar transcrições. A aplicação usa um grupo de usuários do Amazon Cognito para autenticar seus usuários. Os usuários autenticados têm permissões AWS Identity and Access Management (IAM) para acessar os AWS serviços necessários.

Para obter o código-fonte completo e instruções sobre como configurar e executar, veja o exemplo completo em [GitHub](#).

Esse exemplo também está disponível no [Guia do desenvolvedor do AWS SDK for JavaScript v3](#).

Serviços utilizados neste exemplo

- Identidade do Amazon Cognito
- Amazon S3
- Amazon Transcribe

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Criar uma aplicação de transmissão do Amazon Transcribe

O exemplo de código a seguir mostra como construir uma aplicação que registra, transcreve e traduz áudio ao vivo em tempo real, e envia os resultados por e-mail.

JavaScript

SDK para JavaScript (v3)

Mostra como usar o Amazon Transcribe para construir uma aplicação que registra, transcreve e traduz áudio ao vivo em tempo real, e envia os resultados por e-mail usando o Amazon Simple Email Service (Amazon SES).

Para obter o código-fonte completo e instruções sobre como configurar e executar, veja o exemplo completo em [GitHub](#).

Serviços utilizados neste exemplo

- Amazon Comprehend
- Amazon SES
- Amazon Transcribe
- Amazon Translate

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Converta texto em fala e vice-versa em texto usando um AWS SDK

O exemplo de código a seguir mostra como:

- Usar o Amazon Polly para sintetizar um arquivo de entrada de texto simples (UTF-8) para um arquivo de áudio.
- Fazer upload do arquivo de áudio para um bucket do Amazon S3.
- Usar o Amazon Transcribe para converter o arquivo de áudio em texto.
- Exibir o texto.

Rust

SDK para Rust

Use o Amazon Polly para sintetizar um arquivo de texto simples (UTF-8) para um arquivo de áudio, fazer upload do arquivo de áudio para um bucket do Amazon S3, usar o Amazon Transcribe para converter esse arquivo de áudio em texto e exibir o texto.

Para obter o código-fonte completo e instruções sobre como configurar e executar, veja o exemplo completo em [GitHub](#).

Serviços utilizados neste exemplo

- Amazon Polly
- Amazon S3
- Amazon Transcribe

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulte [Usando esse serviço com um AWS SDK](#). Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Segurança em Amazon Transcribe

A segurança para com a nuvem na AWS é a nossa maior prioridade. Como cliente da AWS, você se beneficia de um datacenter e uma arquitetura de rede criados para atender os requisitos da maioria das organizações com exigências de segurança.

A segurança é uma responsabilidade compartilhada entre a AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como segurança da nuvem e segurança na nuvem:

- Segurança da nuvem: AWS é responsável por proteger a infraestrutura que executa AWS serviços da Nuvem AWS. AWS também fornece serviços que você pode usar com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos [Programas de conformidade da AWS](#). Para saber mais sobre os programas de conformidade que se aplicam ao Amazon Transcribe, consulte [Serviços da AWS em escopo por programa de conformidade](#)
- Segurança na nuvem: sua responsabilidade é determinada pelo AWS serviço da que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da sua empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda a entender como aplicar o modelo de responsabilidade compartilhada ao usar o Amazon Transcribe. Os tópicos a seguir mostram como configurar o Amazon Transcribe para atender aos seus objetivos de segurança e conformidade. Saiba também como usar outros AWS serviços da para monitorar e proteger os Amazon Transcribe recursos do.

Tópicos

- [Identity and Access Management para Amazon Transcribe](#)
- [Proteção de dados no Amazon Transcribe](#)
- [Monitoramento Amazon Transcribe](#)
- [Validação de conformidade para Amazon Transcribe](#)
- [Resiliência no Amazon Transcribe](#)
- [Segurança da infraestrutura no Amazon Transcribe](#)
- [Análise e gerenciamento de vulnerabilidades no Amazon Transcribe](#)
- [Melhores práticas de segurança do Amazon Transcribe](#)

Identity and Access Management para Amazon Transcribe

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) a usar Amazon Transcribe os recursos. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

- [Público](#)
- [Autenticando com identidades](#)
- [Gerenciando acesso usando políticas](#)
- [Como Amazon Transcribe funciona com o IAM](#)
- [Prevenção do problema do substituto confuso entre serviços](#)
- [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#)
- [Solução de problemas Amazon Transcribe de identidade e acesso](#)

Público

A forma como você usa AWS Identity and Access Management (IAM) difere, dependendo do trabalho que você faz Amazon Transcribe.

Usuário do serviço — Se você usar o Amazon Transcribe serviço para realizar seu trabalho, seu administrador fornecerá as credenciais e as permissões de que você precisa. À medida que você usa mais Amazon Transcribe recursos para fazer seu trabalho, talvez precise de permissões adicionais. Entender como o acesso é gerenciado pode ajudá-lo a solicitar as permissões corretas ao seu administrador. Se não for possível acessar um atributo no Amazon Transcribe, consulte [Solução de problemas Amazon Transcribe de identidade e acesso](#).

Administrador de serviços — Se você é responsável pelos Amazon Transcribe recursos da sua empresa, provavelmente tem acesso total Amazon Transcribe a. É seu trabalho determinar quais Amazon Transcribe recursos e recursos seus usuários do serviço devem acessar. Assim, você deve enviar solicitações ao administrador do IAM para alterar as permissões dos usuários de seu serviço. Revise as informações nesta página para entender os Introdução ao IAM. Para saber mais sobre como sua empresa pode usar o IAM com Amazon Transcribe, consulte [Como Amazon Transcribe funciona com o IAM](#).

Administrador do IAM: Se você for um administrador do IAM, talvez queira saber detalhes sobre como pode gravar políticas para gerenciar acesso ao Amazon Transcribe. Para ver exemplos de políticas Amazon Transcribe baseadas em identidade que você pode usar no IAM, consulte.

[Exemplos de políticas baseadas em identidade do Amazon Transcribe](#)

Autenticando com identidades

A autenticação é como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado (conectado AWS) como o Usuário raiz da conta da AWS, como usuário do IAM ou assumindo uma função do IAM.

Você pode entrar AWS como uma identidade federada usando credenciais fornecidas por meio de uma fonte de identidade. AWS IAM Identity Center Usuários (IAM Identity Center), a autenticação de login único da sua empresa e suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como identidade federada, o administrador já configurou anteriormente a federação de identidades usando perfis do IAM. Ao acessar AWS usando a federação, você está assumindo indiretamente uma função.

Dependendo do tipo de usuário que você é, você pode entrar no AWS Management Console ou no portal de AWS acesso. Para obter mais informações sobre como fazer login AWS, consulte [Como fazer login Conta da AWS no](#) Guia do Início de Sessão da AWS usuário.

Se você acessar AWS programaticamente, AWS fornece um kit de desenvolvimento de software (SDK) e uma interface de linha de comando (CLI) para assinar criptograficamente suas solicitações usando suas credenciais. Se você não usa AWS ferramentas, você mesmo deve assinar as solicitações. Para obter mais informações sobre como usar o método recomendado para assinar solicitações por conta própria, consulte [Assinatura de solicitações de AWS API](#) no Guia do usuário do IAM.

Independente do método de autenticação usado, também pode ser exigido que você forneça informações adicionais de segurança. Por exemplo, AWS recomenda que você use a autenticação multifator (MFA) para aumentar a segurança da sua conta. Para saber mais, consulte [Autenticação multifator](#) no Guia AWS IAM Identity Center do usuário e [Utilizar a autenticação multifator \(MFA\) na AWS](#) no Guia do usuário do IAM.

Conta da AWS usuário root

Ao criar uma Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos Serviços da AWS os recursos da conta. Essa identidade é chamada de usuário Conta da AWS

raiz e é acessada fazendo login com o endereço de e-mail e a senha que você usou para criar a conta. É altamente recomendável não usar o usuário raiz para tarefas diárias. Proteja as credenciais do usuário raiz e use-as para executar as tarefas que somente ele puder executar. Para obter a lista completa das tarefas que exigem login como usuário raiz, consulte [Tarefas que exigem credenciais de usuário raiz](#) no Guia do Usuário do IAM.

Identidade federada

Como prática recomendada, exija que usuários humanos, incluindo usuários que precisam de acesso de administrador, usem a federação com um provedor de identidade para acessar Serviços da AWS usando credenciais temporárias.

Uma identidade federada é um usuário do seu diretório de usuários corporativo, de um provedor de identidade da web AWS Directory Service, do diretório do Identity Center ou de qualquer usuário que acesse usando credenciais fornecidas Serviços da AWS por meio de uma fonte de identidade. Quando as identidades federadas são acessadas Contas da AWS, elas assumem funções, e as funções fornecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos usar o AWS IAM Identity Center. Você pode criar usuários e grupos no IAM Identity Center ou pode se conectar e sincronizar com um conjunto de usuários e grupos em sua própria fonte de identidade para uso em todos os seus Contas da AWS aplicativos. Para obter mais informações sobre o Centro de Identidade do IAM, consulte [“O que é o Centro de Identidade do IAM?”](#) no Guia do usuário AWS IAM Identity Center .

Usuários e grupos do IAM

Um [usuário do IAM](#) é uma identidade dentro da sua Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, recomendamos contar com credenciais temporárias em vez de criar usuários do IAM com credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de uso específicos que exijam credenciais de longo prazo com usuários do IAM, recomendamos alternar as chaves de acesso. Para obter mais informações, consulte [Altere as chaves de acesso regularmente para casos de uso que exijam credenciais de longo prazo](#) no Guia do Usuário do IAM.

Um [grupo do IAM](#) é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível usar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdmins e conceder a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte [Quando criar um usuário do IAM \(em vez de um perfil\)](#) no Guia do usuário do IAM.

Perfis do IAM

Uma [função do IAM](#) é uma identidade dentro da sua Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. Você pode assumir temporariamente uma função do IAM no AWS Management Console [trocando de funções](#). Você pode assumir uma função chamando uma operação de AWS API AWS CLI ou usando uma URL personalizada. Para obter mais informações sobre métodos para o uso de perfis, consulte [Utilizar perfis do IAM](#) no Guia do usuário do IAM.

Funções do IAM com credenciais temporárias são úteis nas seguintes situações:

- **Acesso de usuário federado:** para atribuir permissões a identidades federadas, você pode criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas pelo mesmo. Para obter mais informações sobre perfis para federação, consulte [Criar um perfil para um provedor de identidades de terceiros](#) no Guia do Usuário do IAM. Se você usar o Centro de identidade do IAM, configure um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o Centro de identidade do IAM correlaciona o conjunto de permissões a um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte [Conjuntos de permissões](#) no Guia do usuário AWS IAM Identity Center .
- **Permissões temporárias para usuários do IAM** — um usuário ou um perfil do IAM pode presumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- **Acesso entre contas** — é possível usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) em outra conta acesse recursos em sua conta. Os perfis são a principal forma de conceder acesso entre contas. No entanto, com alguns Serviços da AWS, você pode anexar uma política diretamente a um recurso (em vez de usar uma função como proxy). Para saber a diferença entre funções e políticas baseadas em recursos para acesso entre contas, consulte [Acesso a recursos entre contas no IAM no Guia do](#) usuário do IAM.
- **Acesso entre serviços** — Alguns Serviços da AWS usam recursos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicativos no Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso

usando as permissões do principal de chamada, usando um perfil de serviço ou um perfil vinculado a serviço.

- Sessões de acesso direto (FAS) — Quando você usa um usuário ou uma função do IAM para realizar ações AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) para fazer solicitações aos serviços posteriores. As solicitações do FAS são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou com recursos para ser concluída. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).
- Função de serviço: um perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do Usuário do IAM.
- Função vinculada ao serviço — Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um AWS service (Serviço da AWS). O serviço pode presumir a função de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para funções vinculadas ao serviço.
- Aplicativos em execução no Amazon EC2 — Você pode usar uma função do IAM para gerenciar credenciais temporárias para aplicativos que estão sendo executados em uma instância do EC2 e fazendo AWS CLI solicitações de API. É preferível fazer isso e armazenar chaves de acesso na instância do EC2. Para atribuir uma AWS função a uma instância do EC2 e disponibilizá-la para todos os seus aplicativos, você cria um perfil de instância anexado à instância. Um perfil de instância contém o perfil e permite que os programas em execução na instância do EC2 obtenham credenciais temporárias. Para mais informações, consulte [Utilizar um perfil do IAM para conceder permissões a aplicações em execução nas instâncias do Amazon EC2](#) no Guia do usuário do IAM.

Para saber se deseja usar perfis do IAM, consulte [Quando criar um perfil do IAM \(em vez de um usuário\)](#) no Guia do usuário do IAM.

Gerenciando acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política é um objeto AWS que, quando associada a uma identidade ou recurso, define suas

permissões. AWS avalia essas políticas quando um principal (usuário, usuário raiz ou sessão de função) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas é armazenada AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte [Visão geral das políticas JSON](#) no Guia do Usuário do IAM.

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Por padrão, usuários e funções não têm permissões. Para conceder aos usuários permissões para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem presumir os perfis.

As políticas do IAM definem permissões para uma ação independente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de função da AWS Management Console AWS CLI, da ou da AWS API.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criando políticas do IAM](#) no Guia do Usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas ainda adicionalmente como políticas em linha ou políticas gerenciadas. As políticas em linha são anexadas diretamente a um único usuário, grupo ou perfil. As políticas gerenciadas são políticas autônomas que você pode associar a vários usuários, grupos e funções em seu Conta da AWS. As políticas AWS gerenciadas incluem políticas gerenciadas e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte [Escolher entre políticas gerenciadas e políticas em linha](#) no Guia do Usuário do IAM.

Políticas baseadas em recursos

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os

administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso e em que condições. Você deve [especificar uma entidade principal](#) em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes as políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

O Amazon S3 e o Amazon VPC são exemplos de serviços que oferecem suporte a ACLs. AWS WAF Para saber mais sobre ACLs, consulte [Visão geral da lista de controle de acesso \(ACL\)](#) no Guia do Desenvolvedor do Amazon Simple Storage Service.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- **Limites de permissões:** um limite de permissões é um atributo avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou perfil do IAM). É possível definir um limite de permissões para uma entidade. As permissões resultantes são a interseção das políticas baseadas em identidade de uma entidade com seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou o perfil no campo `Principal` não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte [Limites de permissões para identidades do IAM](#) no Guia do Usuário do IAM.
- **Políticas de controle de serviço (SCPs)** — SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (OU) em. AWS Organizations AWS Organizations é um serviço para agrupar e gerenciar centralmente várias Contas da AWS que sua empresa possui. Se você habilitar todos os atributos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as contas. O SCP limita as permissões para entidades nas contas dos membros, incluindo cada

uma Usuário raiz da conta da AWS. Para obter mais informações sobre o Organizações e SCPs, consulte [How SCPs work](#) (Como os SCPs funcionam) no Guia do usuário do AWS Organizations .

- Políticas de sessão: são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou do perfil e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em atributo. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte [Políticas de sessão](#) no Guia do Usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte [Lógica de avaliação de políticas](#) no Guia do usuário do IAM.

Como Amazon Transcribe funciona com o IAM

Antes de usar o IAM para gerenciar o acesso Amazon Transcribe, saiba com quais recursos do IAM estão disponíveis para uso Amazon Transcribe.

IAM recursos que você pode usar com Amazon Transcribe

IAM recurso	Amazon Transcribe apoio
Políticas baseadas em identidade	Sim
Políticas baseadas em recursos	Não
Ações das políticas	Sim
Atributos de políticas	Sim
Chaves de condição de política (específicas do serviço)	Sim
ACLs	Não

IAM recurso	Amazon Transcribe apoio
ABAC (tags em políticas)	Parcial
Credenciais temporárias	Sim
Permissões de entidade principal	Sim
Perfis de serviço	Sim
Perfis vinculados ao serviço	Não

Para obter uma visão geral de como Amazon Transcribe e outros AWS serviços funcionam com a maioria dos IAM recursos, consulte [AWS os serviços que funcionam com IAM](#) no Guia do IAM usuário.

Políticas baseadas em identidade para o Amazon Transcribe

Suporta políticas baseadas em identidade	Sim
--	-----

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte [Criando políticas do IAM](#) no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações ou recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Você não pode especificar a entidade principal em uma política baseada em identidade porque ela se aplica ao usuário ou perfil ao qual ela está anexada. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte [Referência de elementos da política JSON do IAM](#) no Guia do Usuário do IAM.

Exemplos de políticas baseadas em identidade para Amazon Transcribe

Para ver exemplos de políticas Amazon Transcribe baseadas em identidade, consulte. [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#)

Políticas baseadas em recursos no Amazon Transcribe

Oferece compatibilidade com políticas baseadas em recursos	Não
--	-----

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso e em que condições. Você deve [especificar uma entidade principal](#) em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Para permitir o acesso entre contas, você pode especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em atributo. Adicionar uma entidade principal entre contas à política baseada em recurso é apenas metade da tarefa de estabelecimento da relação de confiança. Quando o principal e o recurso são diferentes Contas da AWS, um administrador do IAM na conta confiável também deve conceder permissão à entidade principal (usuário ou função) para acessar o recurso. Eles concedem permissão ao anexar uma política baseada em identidade para a entidade. No entanto, se uma política baseada em recurso conceder acesso a uma entidade principal na mesma conta, nenhuma política baseada em identidade adicional será necessária. Para obter mais informações, consulte [Acesso a recursos entre contas no IAM](#) no Guia do usuário do IAM.

Ações de políticas para o Amazon Transcribe

Oferece compatibilidade com ações de políticas	Sim
--	-----

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Action` de uma política JSON descreve as ações que você pode usar para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. Existem algumas exceções, como ações somente de permissão, que não têm

uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista de Amazon Transcribe ações, consulte [Ações definidas por Amazon Transcribe](#) na Referência de Autorização de Serviço.

As ações de política Amazon Transcribe usam o `transcribe` prefixo antes da ação. Para especificar várias ações em uma única instrução, separe-as com vírgulas.

```
"Action": [  
    "transcribe:action1",  
    "transcribe:action2"  
]
```

Você também pode especificar várias ações usando caracteres-curinga (*). Por exemplo, para especificar todas as ações que começam com a palavra `List`, inclua a seguinte ação:

```
"Action": "transcribe:List*"
```

Para ver exemplos de políticas Amazon Transcribe baseadas em identidade, consulte. [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#)

recursos de políticas para Amazon Transcribe

Oferece compatibilidade com recursos de políticas	Sim
---	-----

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática recomendada, especifique um recurso usando seu [nome do recurso da Amazon \(ARN\)](#). Isso pode ser feito para ações que oferecem compatibilidade com um tipo de recurso específico, conhecido como permissões em nível de recurso.

Para ações que não oferecem compatibilidade com permissões em nível de recurso, como operações de listagem, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

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

Para ver uma lista dos tipos de Amazon Transcribe recursos e seus ARNs, consulte [Recursos definidos por Amazon Transcribe](#) na Referência de Autorização de Serviço. Para saber com quais ações é possível especificar o ARN de cada atributo, consulte [Ações definidas pelo Amazon Transcribe](#).

Para ver exemplos de políticas Amazon Transcribe baseadas em identidade, consulte. [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#)

Chaves de condição de política para Amazon Transcribe

Suporta chaves de condição de política específicas de serviço	Sim
---	-----

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Condition` (ou bloco `Condition`) permite que você especifique condições nas quais uma instrução estiver em vigor. O elemento `Condition` é opcional. É possível criar expressões condicionais que usem [agentes de condição](#), como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação.

Se você especificar vários elementos `Condition` em uma instrução ou várias chaves em um único `Condition` elemento, a AWS os avaliará usando uma operação lógica AND. Se você especificar vários valores para uma única chave de condição, AWS avalia a condição usando uma OR operação lógica. Todas as condições devem ser atendidas antes que as permissões da instrução sejam concedidas.

Você também pode usar variáveis de espaço reservado ao especificar condições. Por exemplo, é possível conceder a um usuário do IAM permissão para acessar um atributo somente se ele estiver marcado com seu nome de usuário do IAM. Para obter mais informações, consulte [Elementos da política do IAM: variáveis e tags](#) no Guia do usuário do IAM.

AWS suporta chaves de condição globais e chaves de condição específicas do serviço. Para ver todas as chaves de condição AWS globais, consulte as [chaves de contexto de condição AWS global](#) no Guia do usuário do IAM.

Para ver uma lista de chaves de Amazon Transcribe condição, consulte [Chaves de condição Amazon Transcribe](#) na Referência de autorização de serviço. Para saber com quais ações e recursos você pode usar uma chave de condição, consulte [Ações definidas por Amazon Transcribe](#).

Para ver exemplos de políticas Amazon Transcribe baseadas em identidade, consulte. [Exemplos de políticas baseadas em identidade do Amazon Transcribe](#)

ACLs em Amazon Transcribe

Oferece compatibilidade com ACLs	Não
----------------------------------	-----

As listas de controle de acesso (ACLs) controlam quais entidades principais (membros, usuários ou perfis da conta) têm permissões para acessar um recurso. As ACLs são semelhantes as políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

ABAC com Amazon Transcribe

Oferece compatibilidade com ABAC (tags em políticas)	Parcial
--	---------

O controle de acesso baseado em recurso (ABAC) é uma estratégia de autorização que define permissões com base em recursos. Em AWS, esses atributos são chamados de tags. Você pode anexar tags a entidades do IAM (usuários ou funções) e a vários AWS recursos. A marcação de entidades e atributos é a primeira etapa do ABAC. Em seguida, você cria políticas de ABAC para permitir operações quando a tag da entidade principal corresponder à tag do recurso que ela estiver tentando acessar.

O ABAC é útil em ambientes que estão crescendo rapidamente e ajuda em situações onde o gerenciamento de políticas se torna um problema.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no [elemento de condição](#) de uma política usando as `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou chaves de condição `aws:TagKeys`.

Se um serviço oferecer suporte às três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço oferecer suporte às três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para obter mais informações sobre o ABAC, consulte [O que é ABAC?](#) no Guia do Usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte [Utilizar controle de acesso baseado em atributos \(ABAC\)](#) no Guia do usuário do IAM.

Para obter mais informações sobre a marcação de Amazon Transcribe recursos, consulte [Marcar recursos](#). Para obter informações mais detalhadas sobre o controle de acesso baseado em tags, consulte [Controlar o acesso a recursos da AWS usando tags](#).

Usar credenciais temporárias com o Amazon Transcribe

Oferece compatibilidade com credenciais temporárias	Sim
---	-----

Alguns Serviços da AWS não funcionam quando você faz login usando credenciais temporárias. Para obter informações adicionais, incluindo quais Serviços da AWS funcionam com credenciais temporárias, consulte Serviços da AWS [“Trabalhe com o IAM”](#) no Guia do usuário do IAM.

Você está usando credenciais temporárias se fizer login AWS Management Console usando qualquer método, exceto um nome de usuário e senha. Por exemplo, quando você acessa AWS usando o link de login único (SSO) da sua empresa, esse processo cria automaticamente credenciais temporárias. Você também cria automaticamente credenciais temporárias quando faz login no console como usuário e, em seguida, alterna perfis. Para obter mais informações sobre como alternar funções, consulte [Alternar para um perfil \(console\)](#) no Guia do usuário do IAM.

Você pode criar manualmente credenciais temporárias usando a AWS API AWS CLI ou. Em seguida, você pode usar essas credenciais temporárias para acessar AWS. AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para obter mais informações, consulte [Credenciais de segurança temporárias no IAM](#).

Permissões de entidade principal entre serviços para o Amazon Transcribe

Suporte para o recurso Encaminhamento de sessões de acesso (FAS)	Sim
--	-----

Quando você usa um usuário ou uma função do IAM para realizar ações AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) para fazer solicitações aos serviços posteriores. As solicitações do FAS são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou com recursos para ser concluída. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte [Encaminhar sessões de acesso](#).

Funções de serviço para Amazon Transcribe

Oferece compatibilidade com funções de serviço	Sim
--	-----

O perfil de serviço é um [perfil do IAM](#) que um serviço assume para realizar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte [Criar um perfil para delegar permissões a um AWS service \(Serviço da AWS\)](#) no Guia do Usuário do IAM.

Warning

Alterar as permissões de uma função de serviço pode interromper Amazon Transcribe a funcionalidade. Edite as funções de serviço somente quando Amazon Transcribe fornecer orientação para fazer isso.

Funções vinculadas ao serviço para o Amazon Transcribe

Oferece suporte a perfis vinculados ao serviço	Não
--	-----

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode presumir a função de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para funções vinculadas ao serviço.

Amazon Transcribe não oferece suporte a funções vinculadas a serviços.

Para obter detalhes sobre como criar ou gerenciar funções vinculadas a serviços para outros serviços, consulte [AWS serviços que funcionam](#) com IAM. Encontre um serviço na tabela que inclua um Yes na coluna Função vinculada ao serviço. Escolha o link Sim para visualizar a documentação do perfil vinculado a serviço desse serviço.

Prevenção do problema do substituto confuso entre serviços

Um deputado confuso é uma entidade (um serviço ou uma conta) que é coagida por outra entidade a realizar uma ação. Esse tipo de representação pode acontecer entre contas e serviços.

Para a Conta da AWS Esta seção se concentra na prevenção de delegados confusos entre serviços, específica para Amazon Transcribe; no entanto, você pode aprender mais sobre esse tópico na seção de [problemas de delegados confusos IAM](#) do Guia do Usuário.

Para limitar as permissões IAM concedidas a Amazon Transcribe para acessar seus recursos, recomendamos usar as chaves de contexto de condição global [aws:SourceArn](#) e [aws:SourceAccount](#) em suas políticas de recursos.

Se você utilizar ambas as chaves de contexto de condição global, e o `aws:SourceArn` valor e a condição global contiver o `Conta da AWS ID`, o `aws:SourceAccount` `Conta da AWS` valor e a entrada `aws:SourceArn` deverão utilizar o mesmo `Conta da AWS` quando utilizados na mesma instrução de política.

Use `aws:SourceArn` se quiser que apenas um recurso seja associado ao acesso entre serviços. Se você quiser associar qualquer recurso ao acesso entre serviços, use `aws:SourceAccount`. `Conta da AWS`

Note

A maneira mais eficaz de se proteger do problema do substituto confuso é usar a chave de contexto de condição `aws:SourceArn` global com o ARN completo do recurso. Se você não souber o ARN completo ou se estiver especificando vários recursos, use a chave de condição de contexto `aws:SourceArn` global com curingas (*) para as partes desconhecidas do ARN. Por exemplo, `arn:aws:transcribe::123456789012:*`.

Para obter um exemplo de política de assumir funções que mostra como você pode evitar uma questão confusa de deputados, consulte [Política de prevenção de 'confused deputy'](#).

Exemplos de políticas baseadas em identidade do Amazon Transcribe

Por padrão, usuários e funções não têm permissão para criar ou modificar recursos do Amazon Transcribe. Eles também não podem executar tarefas usando o AWS Management Console, a AWS Command Line Interface (AWS CLI) ou a AWS API. Para dar aos usuários permissão para executar ações nos recursos de que precisam, um administrador do IAM pode criar O administrador pode então adicionar as políticas do IAM às funções, e os usuários podem assumir as funções.

Para saber como criar uma política baseada em identidade do IAM usando esses exemplos de documento de política JSON, consulte [Criação de políticas do IAM](#) no Guia do usuário do IAM.

Para obter detalhes sobre ações e tipos de recurso definidos pelo Amazon Transcribe, por exemplo, o formato dos ARNs para cada um dos tipos de recurso, consulte [Ações, recursos e chaves de condição do Amazon Transcribe](#) na Referência de autorização do serviço.

Tópicos

- [Práticas recomendadas de políticas](#)
- [Como usar a AWS Management Console](#)
- [Permissões para IAM funções do](#)
- [Permissões necessárias para chaves Amazon S3 de criptografia](#)
- [Permitir que os usuários visualizem suas próprias permissões](#)
- [AWS KMS política de contexto de criptografia](#)
- [Política de prevenção de 'confused deputy'](#)
- [Visualização de trabalhos de transcrição com base em tags](#)

Práticas recomendadas de políticas

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do Amazon Transcribe em sua conta. Essas ações podem incorrer em custos para a Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas gerenciadas pela AWS e avance para as permissões de privilégio mínimo: para começar a conceder permissões a seus usuários e workloads, use as políticas gerenciadas pela AWS que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis na sua Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo cliente da AWS específicas para seus casos de uso. Para

obter mais informações, consulte [Políticas gerenciadas pela AWS](#) ou [Políticas gerenciadas pela AWS para funções de trabalho](#) no Guia do usuário do IAM.

- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre como usar o IAM para aplicar permissões, consulte [Políticas e permissões no IAM](#) no Guia do usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso: você pode adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, você pode escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso a ações de serviço, se elas forem usadas por meio de um AWS service (Serviço da AWS) específico, como o AWS CloudFormation. Para obter mais informações, consulte [Elementos de política JSON do IAM: Condition](#) no Manual do usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de 100 verificações de política e recomendações acionáveis para ajudar você a criar políticas seguras e funcionais. Para obter mais informações, consulte [Validação de políticas do IAM Access Analyzer](#) no Guia do usuário do IAM.
- Require multi-factor authentication (MFA) (Exigir autenticação multifator (MFA)): se houver um cenário que exija usuários do IAM ou um usuário raiz em sua Conta da AWS, ative a MFA para obter segurança adicional. Para exigir a MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter mais informações, consulte [Configuração de acesso à API protegido por MFA](#) no Guia do usuário do IAM.

Para obter mais informações sobre as práticas recomendadas do IAM, consulte [Práticas recomendadas de segurança no IAM](#) no Guia do usuário do IAM.

Como usar a AWS Management Console

Para acessar o console da Amazon Transcribe, você deve ter um conjunto mínimo de permissões. Essas permissões dão autorização para que você liste e visualize detalhes sobre os Amazon Transcribe recursos do Conta da AWS. Se você criar uma política baseada em identidade que seja mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou perfis) com essa política.

Não é necessário conceder permissões mínimas do console para usuários que fazem chamadas somente à AWS CLI ou à API do AWS. Em vez disso, permita o acesso somente às ações que correspondem à operação da API que estão tentando executar.

Para garantir que uma entidade (usuários e funções) possa usar o [AWS Management Console](#), anexe uma das seguintes políticas AWS gerenciadas a ela.

- `AmazonTranscribeFullAccess`: concede acesso total para criar, ler, atualizar, excluir e executar todos os Amazon Transcribe recursos. Ele também permite o acesso a Amazon S3 buckets com `transcribe` o nome do bucket.
- `AmazonTranscribeReadOnlyAccess`: concede acesso somente de leitura aos Amazon Transcribe recursos para que você possa obter e listar trabalhos de transcrição e vocabulários personalizados.

Note

É possível analisar as políticas de permissões gerenciadas pela fazendo login IAM AWS Management Console e pesquisando pelo nome da política. Uma pesquisa por “transcrever” retorna as duas políticas listadas acima (`AmazonTranscribeReadOnlyAccess` e `AmazonTranscribeFullAccess`).

Além disso, você pode criar políticas personalizadas do IAM para conceder permissões a ações da API do Amazon Transcribe. É possível anexar essas

Permissões para IAM funções do

Se você criar uma IAM função para chamar Amazon Transcribe, ela deverá ter permissão para acessar o Amazon S3 bucket. Se aplicável, o também KMS key deve ser usado para criptografar o conteúdo do bucket. Consulte as seções a seguir para ver exemplos de políticas.

Políticas de confiança

A IAM entidade que você usa para fazer sua solicitação de transcrição deve ter uma política de confiança que permita Amazon Transcribe assumir essa função. Use a política de Amazon Transcribe confiança a seguir. Observe que, se você estiver fazendo uma solicitação de análise de chamadas em tempo real com a análise pós-chamada ativada, deverá usar a “Política de confiança para análise de chamadas em tempo real”.

Política de confiança para Amazon Transcribe

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "transcribe.amazonaws.com"
        ]
      },
      "Action": [
        "sts:AssumeRole"
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "StringLike": {
          "aws:SourceArn": "arn:aws:transcribe:us-west-2:111122223333:*"
        }
      }
    }
  ]
}
```

Política de confiança para análise de chamadas em tempo real

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

```

    "StringEquals": {
      "aws:SourceAccount": "111122223333"
    },
    "StringLike": {
      "aws:SourceArn": "arn:aws:transcribe:us-west-2:111122223333:*"
    }
  }
}
]
}

```

Amazon S3 política de compartimento de entrada

A política a seguir concede a uma IAM função permissão para acessar arquivos do bucket de entrada especificado.

```

{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "s3:GetObject",
      "s3:ListBucket"
    ],
    "Resource": [
      "arn:aws:s3:::DOC-EXAMPLE-INPUT-BUCKET",
      "arn:aws:s3:::DOC-EXAMPLE-INPUT-BUCKET/*"
    ]
  }
}

```

Amazon S3 política de bucket de saída

A política a seguir concede a uma IAM função permissão para gravar arquivos no bucket de saída especificado.

```

{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "s3:PutObject"
    ]
  }
}

```

```

    ],
    "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-OUTPUT-BUCKET/*"
    ]
}
}

```

Permissões necessárias para chaves Amazon S3 de criptografia

Se você estiver usando um KMS key para criptografar um Amazon S3 bucket, inclua o seguinte na KMS key política. Isso fornece ao Amazon Transcribe acesso ao conteúdo do bucket. Para obter mais informações sobre como permitir o acesso a KMS keys, consulte [Permitir o Contas da AWS acesso externo](#) a KMS key no Guia do AWS KMS desenvolvedor.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
      },
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:us-west-2:111122223333:key/KMS-Example-KeyId"
    }
  ]
}

```

Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como você pode criar uma política que permite que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou de forma programática usando a AWS CLI ou a API da AWS.

```

{
  "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": "*"
}
]
}

```

AWS KMS política de contexto de criptografia

A política a seguir concede à IAM função "ExampleRole" permissão para usar as operações AWS KMS Decrypt e Encrypt para esse específico KMS key. Essa política funciona somente para solicitações com pelo menos um par de contexto de criptografia, nesse caso "color:indigoBlue". Para obter mais informações sobre o contexto de AWS KMS criptografia, consulte [Contexto de criptografia do AWS KMS](#).

```

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

```



```

    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": [
    "kms:Decrypt",
    "kms:DescribeKey",
    "kms:Encrypt",
    "kms:GenerateDataKey*",
    "kms:ReEncrypt*"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:color": "indigoBlue"
    }
  }
}

```

Política de prevenção de 'confused deputy'

Aqui está um exemplo de uma política de assumir funções que mostra como você pode usar `aws:SourceArn` e `aws:SourceAccount` para evitar um problema confuso de deputados. Para obter mais informações sobre prevenção de delegados confusos, consulte [Prevenção do problema do substituto confuso entre serviços](#).

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "transcribe.amazonaws.com"
      },
      "Action": [
        "sts:AssumeRole",
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "StringLike": {
          "aws:SourceArn": "arn:aws:transcribe:us-west-2:111122223333:*"
        }
      }
    }
  ]
}

```

```

    }
  }
}
]
}

```

Visualização de trabalhos de transcrição com base em tags

Você pode usar condições em sua política baseada em identidade para controlar o acesso aos recursos do Amazon Transcribe com base em tags. Este exemplo mostra como criar uma política que permita visualizar uma tarefa de transcrição. No entanto, a permissão será concedida somente se a tag de trabalho de transcrição `owner` tiver o valor do nome de usuário desse usuário. Esta política também concede as permissões necessárias para concluir essa ação usando o [AWS Management Console](#).

Você pode anexar essa política às IAM entidades da sua conta. Se uma função chamada `test-role` tentar visualizar uma tarefa de transcrição, a tarefa de transcrição deverá ser marcada `owner=test-role` ou `owner=test-role` (os nomes das chaves de condição não diferenciam maiúsculas de minúsculas), caso contrário, o acesso será negado. Para obter mais informações, consulte [Elementos da política IAM JSONIAM](#) do

Para obter mais informações sobre como marcar o Amazon Transcribe, consulte [Marcar recursos](#).

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ListTranscriptionJobsInConsole",
      "Effect": "Allow",
      "Action": "transcribe:ListTranscriptionJobs",
      "Resource": "*"
    },
    {
      "Sid": "ViewTranscriptionJobsIfOwner",
      "Effect": "Allow",
      "Action": "transcribe:GetTranscriptionJobs",
      "Resource": "arn:aws:transcribe:*:*:transcription-job/*",
      "Condition": {
        "StringEquals": {"aws:ResourceTag/Owner": "${aws:username}"}
      }
    }
  ]
}

```

```
]
}
```

Solução de problemas Amazon Transcribe de identidade e acesso

Use as informações a seguir para diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com Amazon Transcribe e AWS Identity and Access Management (IAM).

Tópicos

- [Não tenho autorização para executar uma ação no Amazon Transcribe](#)
- [Não estou autorizado a realizar iam: PassRole](#)
- [Quero permitir que pessoas fora da minha Conta da AWS acessem meus Amazon Transcribe recursos](#)

Não tenho autorização para executar uma ação no Amazon Transcribe

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, suas políticas deverão ser atualizadas para permitir que você realize a ação.

O erro do exemplo a seguir ocorre quando o usuário do IAM mateojackson tenta usar o console para visualizar detalhes sobre um atributo *my-example-widget* fictício, mas não tem as permissões `transcribe:GetWidget` fictícias.

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

Nesse caso, a política do usuário mateojackson deve ser atualizada para permitir o acesso ao recurso *my-example-widget* usando a ação `transcribe:GetWidget`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a realizar iam: PassRole

Se você receber uma mensagem de erro informando que não está autorizado a executar a ação `iam:PassRole`, as suas políticas devem ser atualizadas para permitir que você passe uma função para o Amazon Transcribe.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazê-lo, você deve ter permissões para passar o perfil para o serviço.

O exemplo de erro a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta utilizar o console para executar uma ação no Amazon Transcribe. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

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

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que pessoas fora da minha Conta da AWS acessem meus Amazon Transcribe recursos

Você pode criar um perfil que os usuários de outras contas ou pessoas fora da sua organização podem usar para acessar seus recursos. Você pode especificar quem é confiável para assumir o perfil. Para serviços que oferecem compatibilidade com políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se é Amazon Transcribe compatível com esses recursos, consulte [Como Amazon Transcribe funciona com o IAM](#).
- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte [Como fornecer acesso a um usuário do IAM em outro Conta da AWS que você possui](#) no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte [Como fornecer acesso Contas da AWS a terceiros](#) no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte [Conceder acesso a usuários autenticados externamente \(federação de identidades\)](#) no Guia do usuário do IAM.

- Para saber a diferença entre usar funções e políticas baseadas em recursos para acesso entre contas, consulte Acesso a [recursos entre contas no IAM no Guia do](#) usuário do IAM.

Proteção de dados no Amazon Transcribe

O AWS [modelo de responsabilidade compartilhada](#) se aplica à proteção de dados no Amazon Transcribe. Conforme descrito nesse modelo, a AWS é responsável por proteger a infraestrutura global que executa toda a Nuvem AWS. Você é responsável por manter o controle sobre seu conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para obter mais informações sobre a privacidade de dados, consulte as [Perguntas frequentes sobre privacidade de dados](#). Para mais informações sobre a proteção de dados na Europa, consulte o artigo [AWS Shared Responsibility Model and GDPR](#) no Blog de segurança da AWS.

Para fins de proteção de dados, recomendamos que você proteja as Conta da AWS credenciais da e configure as contas de usuário individuais com o AWS IAM Identity Center ou o AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA [multi-factor authentication]) com cada conta.
- Use SSL/TLS para se comunicar com os atributos da AWS. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure o registro em log das atividades da API e do usuário com o .AWS CloudTrail
- Use AWS as soluções de criptografia da , juntamente com todos os controles de segurança padrão dos Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-2 ao acessar a AWS por meio de uma interface de linha de comandos ou uma API, use um endpoint do FIPS. Para ter mais informações sobre endpoints do FIPS, consulte [Federal Information Processing Standard \(FIPS\) 140-2](#).

É altamente recomendável que nunca sejam colocadas informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Nome. Isso inclui trabalhar com a Amazon Transcribe ou outros Serviços da AWS usando

o console, a API, a AWS CLI ou os AWS SDKs. Quaisquer dados inseridos em tags ou campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, recomendamos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

Privacidade do tráfego entre redes

Um Amazon Virtual Private Cloud (Amazon VPC) endpoint para o Amazon Transcribe é uma entidade lógica dentro de uma VPC que permite conectividade somente com o Amazon Transcribe. A Amazon VPC encaminha solicitações para o Amazon Transcribe e encaminha respostas de volta para a VPC. Para obter mais informações, consulte [Conceitos do AWS PrivateLink](#). Para obter mais informações sobre como usar Amazon VPC endpoints com o Amazon Transcribe, consulte [Amazon Transcribe e VPC endpoints de interface \(AWS PrivateLink\)](#).

Criptografia de dados

Criptografia de dados se refere à proteção de dados em trânsito e em repouso. Você pode proteger seus dados usando chaves Amazon S3 gerenciadas ou KMS keys em repouso, junto com o Transport Layer Security (TLS) padrão durante o transporte.

Criptografia em repouso

Amazon Transcribe usa a Amazon S3 chave padrão (SSE-S3) para criptografia no lado do servidor das transcrições colocadas em seu Amazon S3 bucket.

Ao usar a [StartTranscriptionJob](#) operação, você pode especificar a sua própria KMS key para criptografar a saída de um trabalho de transcrição.

O Amazon Transcribe usa um volume do Amazon EBS criptografado com a chave padrão.

Criptografia em trânsito

O Amazon Transcribe usa TLS 1.2 com certificados da AWS para criptografar dados em trânsito. Isso inclui transcrições em streaming.

Gerenciamento de chaves

Amazon Transcribe trabalha com KMS keys para fornecer criptografia aprimorada para seus dados. Com Amazon S3, você pode criptografar sua mídia de entrada ao criar um

trabalho de transcrição. A integração com AWS KMS permite a criptografia da saída de uma [StartTranscriptionJob](#) solicitação.

Se você não especificar uma KMS key, a saída da tarefa de transcrição será criptografada com a Amazon S3 chave padrão (SSE-S3).

Para obter mais informações sobre AWS KMS, consulte o [Guia do AWS Key Management Service desenvolvedor](#).

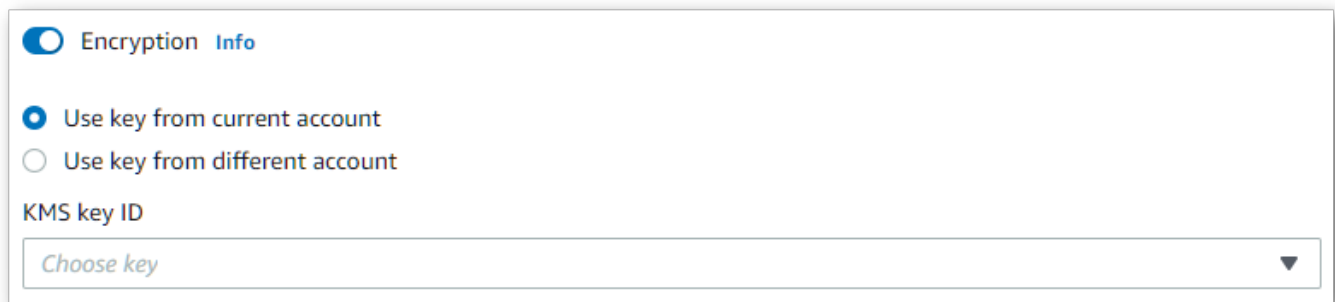
Gerenciamento de chaves usando o AWS Management Console

Para criptografar a saída do seu trabalho de transcrição, você pode escolher entre usar uma KMS key para quem está fazendo a solicitação ou uma KMS key de outra Conta da AWS. Conta da AWS

Se você não especificar uma KMS key, a saída da tarefa de transcrição será criptografada com a Amazon S3 chave padrão (SSE-S3).

Para ativar a criptografia de saída:

1. Em Output data (Dados de saída), escolha Encryption (Criptografia).



The screenshot shows the 'Encryption' section in the AWS console. It features a blue toggle switch labeled 'Encryption Info'. Below it are two radio button options: 'Use key from current account' (which is selected) and 'Use key from different account'. Underneath these options is a text input field labeled 'KMS key ID' with a dropdown arrow and the placeholder text 'Choose key'.

2. Escolha se a KMS key é da Conta da AWS que você está usando atualmente ou de uma diferente Conta da AWS. Se você quiser usar uma chave da atual Conta da AWS, escolha a chave do KMS key ID. Se você estiver usando uma chave de outra Conta da AWS, deverá inserir o ARN da chave. Para usar uma chave de outra Conta da AWS, o chamador deve ter `kms:Encrypt` permissões para a KMS key. Consulte [Criação de uma política chave](#) para obter mais informações.

Gerenciamento de chaves usando a API

Para usar a criptografia de saída com a API, você deve especificar o `KMSKeyId` parâmetro da [StartTranscriptionJob](#) operação [StartCallAnalyticsJob](#) [StartMedicalTranscriptionJob](#), ou.

Se estiver usando uma chave localizada na atualConta da AWS, você pode especificar suaKMS key de uma das quatro maneiras:

1. Use oKMS key ID em si. Por exemplo, `1234abcd-12ab-34cd-56ef-1234567890ab`.
2. Use um alias para oKMS key ID. Por exemplo, `alias/ExampleAlias`.
3. Use o nome do recurso da Amazon (ARN) para aKMS key ID. Por exemplo, `arn:aws:kms:region:account-ID:key/1234abcd-12ab-34cd-56ef-1234567890ab`.
4. Use o ARN para oKMS key alias. Por exemplo, `arn:aws:kms:region:account-ID:alias/ExampleAlias`.

Se estiver usando uma chave localizada em umaConta da AWS posição diferente da atualConta da AWS, você pode especificar suaKMS key de duas maneiras:

1. Use o ARN para oKMS key ID. Por exemplo, `arn:aws:kms:region:account-ID:key/1234abcd-12ab-34cd-56ef-1234567890ab`.
2. Use o ARN para oKMS key alias. Por exemplo, `arn:aws:kms:region:account-ID:alias/ExampleAlias`.

Observe que a entidade que está fazendo a solicitação deve ter permissão para usar o especificadoKMS key.

Contexto de criptografia do AWS KMS

AWS KMSo contexto de criptografia é um mapa de pares chave:valor não secretos e em texto simples. Esse mapa representa dados autenticados adicionais, conhecidos como pares de contexto de criptografia, que fornecem uma camada adicional de segurança para seus dados. Amazon Transcribeexige uma chave de criptografia simétrica para criptografar a saída da transcrição em umAmazon S3 bucket especificado pelo cliente. Para saber mais, consulte [Chaves assimétricas emAWS KMS](#).

Ao criar seus pares de contexto de criptografia, não inclua informações confidenciais. O contexto de criptografia não é secreto — é visível em texto simples em seusCloudTrail Logs do, para você possa usá-lo para identificar e categorizar suas operações de criptografia.

Seu par de contexto de criptografia pode incluir caracteres especiais, como sublinhados (`_`), traços (`-`), barras (`/`, `\`) e dois pontos (`:`).

Tip

Pode ser útil relacionar os valores em seu par de contexto de criptografia aos dados que estão sendo criptografados. Embora não seja obrigatório, recomendamos que você use metadados não confidenciais relacionados ao seu conteúdo criptografado, como nomes de arquivos, valores de cabeçalho ou campos de banco de dados não criptografados.

Para usar a criptografia de saída com a API, defina o `KMSEncryptionContext` parâmetro na [StartTranscriptionJob](#) operação. Para fornecer contexto de criptografia para a operação de criptografia de saída, o `OutputEncryptionKMSKeyId` parâmetro deve fazer referência a uma KMS key ID simétrica.

É possível usar [chaves de AWS KMS condição](#) com IAM políticas para controlar o acesso a uma criptografia simétrica KMS key com base no contexto de criptografia usado na solicitação de uma [operação criptográfica](#). Para ver um exemplo de política de contexto de criptografia, consulte [AWS KMS política de contexto de criptografia](#).

Usar o contexto de criptografia é opcional, mas recomendado. Para obter mais informações, consulte [Contexto de criptografia](#).

Optar por não usar seus dados para melhorar o serviço

Por padrão, Amazon Transcribe armazena e usa as entradas de voz processadas para desenvolver o serviço e melhorar continuamente sua experiência. Você pode optar por não usar seu conteúdo para desenvolvimento e aprimoramento Amazon Transcribe usando uma política AWS Organizations de exclusão. Para obter informações sobre como exclusão dos serviços de [IA, consulte políticas de exclusão dos serviços](#) de IA.

Monitoramento Amazon Transcribe

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho de Amazon Transcribe suas outras AWS soluções. AWS fornece as seguintes ferramentas de monitoramento para observar Amazon Transcribe, relatar quando algo está errado e realizar ações automáticas quando apropriado:

- Amazon CloudWatch monitora seus AWS recursos e os aplicativos que você executa AWS em tempo real. É possível coletar e rastrear métricas, criar painéis personalizados e definir alarmes

que o notificam ou que realizam ações quando uma métrica especificada atinge um limite definido. Por exemplo, você pode CloudWatch monitorar o uso da CPU ou outras métricas em suas Amazon EC2 instâncias e iniciar automaticamente novas instâncias quando necessário.

- Amazon CloudWatch Logspode monitorar, armazenar e acessar seus arquivos de log de Amazon EC2 instâncias e outras fontes. CloudTrail CloudWatch Logs pode monitorar as informações nos arquivos de log e notificá-lo quando determinados limites forem atingidos. É possível também arquivar seus dados de log em armazenamento resiliente.
- AWS CloudTrailcaptura chamadas de API e eventos relacionados feitos por você ou em seu nome Conta da AWS e entrega os arquivos de log em um Amazon S3 bucket especificado por você. Você pode identificar quais usuários e contas ligaram AWS, o endereço IP de origem a partir do qual as chamadas foram feitas e quando elas ocorreram.

Para mais informações, consulte o Guia do usuário do [Amazon CloudWatch](#).

Amazon EventBridgeé um serviço sem servidor que usa eventos para conectar componentes do aplicativo, facilitando a criação de aplicativos escaláveis orientados por eventos. EventBridge fornece um fluxo de dados em tempo real de seus próprios aplicativos, aplicativos de software como serviço (SaaS) e AWS serviços e encaminha esses dados para destinos como. Lambda Você pode monitorar os eventos que ocorrem nos serviços e criar arquiteturas orientadas por eventos. Para mais informações, consulte o Guia do usuário do [Amazon EventBridge](#).

Tópicos

- [Monitoramento Amazon Transcribe com Amazon CloudWatch](#)
- [Monitoramento Amazon Transcribe com AWS CloudTrail](#)
- [Usando Amazon EventBridge com Amazon Transcribe](#)

Monitoramento Amazon Transcribe com Amazon CloudWatch

Você pode monitorar Amazon Transcribe o uso CloudWatch, que coleta dados brutos e os processa em métricas legíveis e quase em tempo real. Essas estatísticas são mantidas por 15 meses, de maneira que você possa acessar informações históricas e ter uma perspectiva melhor de como o aplicativo web ou o serviço está se saindo. Você também pode definir alarmes que observam determinados limites e enviam notificações ou realizam ações quando esses limites são atingidos. Para mais informações, consulte o [Guia do usuário do CloudWatch](#) .

Usando Amazon CloudWatch métricas e dimensões com Amazon Transcribe

Amazon Transcribe oferece suporte a CloudWatch métricas e dimensões, que são dados que podem ajudar você a monitorar o desempenho. As categorias de métricas aceitas incluem tráfego, erros, transferência de dados e latência associados às tarefas de transcrição. As métricas suportadas estão localizadas CloudWatch no namespace AWS/Transcribe.

Note

CloudWatch as métricas de monitoramento são gratuitas e não contam para as cotas CloudWatch de serviço.

Para obter mais informações sobre CloudWatch métricas, consulte Como [usar Amazon CloudWatch métricas](#).

Monitoramento Amazon Transcribe com AWS CloudTrail

Amazon Transcribe é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas Amazon Transcribe por um AWS Identity and Access Management (IAM) usuário ou função, ou por um AWS serviço. CloudTrail captura todas as chamadas de API para Amazon Transcribe. Isso inclui chamadas do AWS Management Console e chamadas de código para as Amazon Transcribe APIs, como eventos. Ao criar uma trilha, você pode permitir a entrega contínua de CloudTrail eventos, incluindo eventos para Amazon Transcribe, em um Amazon S3 bucket. Se você não criar uma trilha, mesmo assim poderá visualizar os eventos mais recentes no AWS Management Console do CloudTrail em Histórico de eventos. Usando as informações coletadas por CloudTrail, você pode ver cada solicitação feita Amazon Transcribe, o endereço IP do qual a solicitação é feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais sobre isso CloudTrail, consulte o [Guia AWS CloudTrail do usuário](#).

Amazon Transcribe e CloudTrail

CloudTrail é ativado no seu Conta da AWS quando você cria a conta. Quando a atividade ocorre em Amazon Transcribe, essa atividade é registrada em um CloudTrail evento junto com outros AWS service (Serviço da AWS) eventos no histórico de CloudTrail eventos. Você pode visualizar, pesquisar e baixar eventos recentes no seu Conta da AWS. Para obter mais informações, consulte [Visualizar eventos com o histórico de eventos do CloudTrail](#).

Para obter um registro contínuo dos eventos em sua Conta da AWS, incluindo eventos para Amazon Transcribe, crie uma trilha. Uma trilha é uma configuração que permite CloudTrail entregar eventos como arquivos de log para um Amazon S3 bucket especificado. CloudTrail os arquivos de log contêm uma ou mais entradas de log. Um evento representa uma solicitação única de qualquer fonte. Ele inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

Por padrão, quando você cria uma trilha no AWS Management Console, a trilha se aplica a todos Regiões da AWS. A trilha registra eventos de todos Regiões da AWS na AWS partição e entrega os arquivos de log ao Amazon S3 bucket que você especificar. Além disso, você pode configurar outros Serviços da AWS para analisar e agir com base nos dados do evento coletados nos CloudTrail registros. Para obter mais informações, consulte:

- [Visão geral da criação de uma trilha](#)
- [CloudTrail Serviços e integrações compatíveis](#)
- [Configurando Amazon SNS notificações para CloudTrail](#)
- [Receber arquivos de log do CloudTrail de várias regiões](#) e [receber arquivos de log do CloudTrail de várias contas](#)

CloudTrail registra todas Amazon Transcribe as ações, que estão documentadas na [Referência da API](#). Por exemplo, as [StartTranscriptionJob](#) operações [CreateVocabularyGetTranscriptionJob](#), e geram entradas nos arquivos de CloudTrail log.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações ajudam a identificar:

- Se a solicitação é feita com credenciais raiz ou de IAM usuário
- Se a solicitação é feita com credenciais de segurança temporárias para um perfil do IAM ou usuário federado.
- Se a solicitação é feita por outro AWS service (Serviço da AWS)

Para obter mais informações, consulte [Elemento userIdentity do CloudTrail](#).

Você também pode agregar arquivos de Amazon Transcribe log de vários Regiões da AWS e vários Contas da AWS em um único Amazon S3 bucket. Para obter mais informações, consulte [Recebendo](#)

[arquivos de CloudTrail log de várias regiões](#) e [Recebendo arquivos de CloudTrail log de várias contas](#).

Exemplo: entradas do arquivo de Amazon Transcribe log

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um Amazon S3 bucket especificado. CloudTrail os arquivos de log contêm uma ou mais entradas de log. Um evento representa uma solicitação única de qualquer fonte. Inclui informações sobre a ação solicitada, como data e hora da ação e parâmetros da solicitação. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

As chamadas para as operações de API [StartTranscriptionJob](#) e [GetTranscriptionJob](#) criam a entrada a seguir.

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "111122223333",
        "arn": "arn:aws:iam:us-west-2:111122223333:user/my-user-name",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "my-user-name"
      },
      "eventTime": "2022-03-07T15:03:45Z",
      "eventSource": "transcribe.amazonaws.com",
      "eventName": "StartTranscriptionJob",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "127.0.0.1",
      "userAgent": "[]",
      "requestParameters": {
        "mediaFormat": "flac",
        "languageCode": "en-US",
        "transcriptionJobName": "my-first-transcription-job",
        "media": {
          "mediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-media-file.flac"
        }
      },
      "responseElements": {
        "transcriptionJob": {
```

```

        "transcriptionJobStatus": "IN_PROGRESS",
        "mediaFormat": "flac",
        "creationTime": "2022-03-07T15:03:44.229000-08:00",
        "transcriptionJobName": "my-first-transcription-job",
        "languageCode": "en-US",
        "media": {
            "mediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-media-file.flac"
        }
    },
    "requestID": "47B8E8D397DCE7A6",
    "eventID": "cdc4b7ed-e171-4cef-975a-ad829d4123e8",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
},
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "111122223333",
        "arn": "arn:aws:iam:us-west-2:111122223333:user/my-user-name",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "my-user-name"
    },
    "eventTime": "2022-03-07T15:07:11Z",
    "eventSource": "transcribe.amazonaws.com",
    "eventName": "GetTranscriptionJob",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "[]",
    "requestParameters": {
        "transcriptionJobName": "my-first-transcription-job"
    },
    "responseElements": {
        "transcriptionJob": {
            "settings": {

            },
            "transcriptionJobStatus": "COMPLETED",
            "mediaFormat": "flac",
            "creationTime": "2022-03-07T15:03:44.229000-08:00",
            "transcriptionJobName": "my-first-transcription-job",
            "languageCode": "en-US",

```

```
        "media": {
            "mediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-media-file.flac"
        },
        "transcript": {
            "transcriptFileUri": "s3://DOC-EXAMPLE-BUCKET/my-first-
transcription-job.json"
        }
    },
    "requestID": "BD8798EACDD16751",
    "eventID": "607b9532-1423-41c7-b048-ec2641693c47",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
]
```

Usando Amazon EventBridge com Amazon Transcribe

Com Amazon EventBridge, você pode responder às mudanças de estado em seus Amazon Transcribe trabalhos iniciando eventos em outros Serviços da AWS. Quando um trabalho de transcrição muda de estado, envia EventBridge automaticamente um evento para um fluxo de eventos. Você pode criar regras que definem os eventos que você deseja monitorar no fluxo de eventos, e a ação que o EventBridge deve executar quando esses eventos ocorrerem. Por exemplo, rotear o evento para outro serviço (ou destino), que pode executar uma ação. Você pode, por exemplo, configurar uma regra para rotear um evento para uma AWS Lambda função quando um trabalho de transcrição for concluído com êxito. Para definir [regras do EventBridge](#), consulte as seções a seguir.

É possível receber notificações de eventos por meio de vários canais, incluindo e-mail, notificações de chat do [AWS Chatbot](#) ou notificações por push do [AWS Console Mobile Application](#). Você também pode ver as notificações na [Central de notificações do console](#). Se você quiser configurar notificações, você pode usar [Notificações de Usuários da AWS](#). Notificações de Usuários da AWS suporta agregação, o que pode reduzir o número de notificações que você recebe durante eventos específicos.

Definindo EventBridge regras

Para definir EventBridge regras, use [AWS Management Console](#). Ao definir uma regra, use o Amazon Transcribe como o nome do serviço. Para ver um exemplo de como criar uma EventBridge regra, consulte [Amazon EventBridge regras](#).

Antes de usar EventBridge, observe as seguintes definições:

- **Evento:** um evento indica uma alteração no estado de um dos trabalhos de transcrição. Por exemplo, quando o `TranscriptionJobStatus` de um trabalho muda de `IN_PROGRESS` para `COMPLETED`.
- **Destino:** um destino é outro AWS service (Serviço da AWS) que processa um evento. Por exemplo, AWS Lambda ou Amazon Simple Notification Service (Amazon SNS). Um destino recebe eventos no formato JSON.
- **Regra** — Uma regra combina os eventos recebidos que você EventBridge deseja observar e os encaminha para um alvo ou alvos para processamento. Se uma regra rotear um evento para vários destinos, os destinos processarão o evento em paralelo. Uma regra pode personalizar o JSON enviado ao destino.

Amazon EventBridge os eventos são emitidos com base no melhor esforço. Para obter mais informações sobre como criar e gerenciar eventos em EventBridge, consulte [Amazon EventBridge eventos](#) no Guia Amazon EventBridge do usuário.

Veja a seguir um exemplo de uma EventBridge regra Amazon Transcribe que é iniciada quando o status de um trabalho de transcrição muda para `COMPLETED` ou `FAILED`.

```
{
  "source": [
    "aws.transcribe"
  ],
  "detail-type": [
    "Transcribe Job State Change"
  ],
  "detail": {
    "TranscriptionJobStatus": [
      "COMPLETED",
      "FAILED"
    ]
  }
}
```

A regra contém os seguintes campos:

- **source:** a origem do evento. Pois Amazon Transcribe, isso é sempre `aws.transcribe`.

- `detail-type`: um identificador para os detalhes do evento. Para o Amazon Transcribe, isso é sempre `Transcribe Job State Change`.
- `detail`: o novo status do trabalho de transcrição. Neste exemplo, a regra inicia um evento quando o status do trabalho muda para `COMPLETED` ou `FAILED`.

Amazon Transcribe eventos

Amazon EventBridge registra vários Amazon Transcribe eventos:

- [Eventos de trabalho de transcrição](#)
- [Eventos de identificação de idiomas](#)
- [Eventos do Call Analytics](#)
- [Eventos pós-chamada do Call Analytics](#)
- [Eventos de vocabulário](#)

Todos esses eventos contêm os seguintes campos compartilhados:

- `version`: a versão dos dados do evento. Esse valor é sempre `0`.
- `id`: Um identificador exclusivo gerado por EventBridge para o evento.
- `detail-type`: um identificador para os detalhes do evento. Por exemplo, `Transcribe Job State Change`.
- `source`: a origem do evento. Pois Amazon Transcribe isso é sempre `aws.transcribe`.
- `account`: o Conta da AWS ID da conta que gerou a chamada da API.
- `time`: a data e a hora em que o evento é entregue.
- `region`: O Região da AWS em que a solicitação é feita.
- `resources` os recursos usados pela chamada de API. Pois Amazon Transcribe, esse campo está sempre vazio.
- `detail`: detalhes adicionais sobre o evento.
 - `FailureReason`: esse campo estará presente se o estado ou status mudar para `FAILED` e descreve o motivo do estado ou status `FAILED`.
 - Cada tipo de evento tem outros campos exclusivos que são exibidos em `detail`. Esses campos exclusivos são definidos nas seções a seguir após cada exemplo de evento.

Eventos de trabalho de transcrição

Quando o estado de um trabalho muda de IN_PROGRESS para COMPLETED ou FAILED, Amazon Transcribe gera um evento. Para identificar o trabalho que mudou de estado e iniciar o evento no destino, use o campo `TranscriptionJobName` do evento. Um Amazon Transcribe evento contém as seguintes informações. Um campo `FailureReason` será adicionado em `detail` se o status do trabalho de transcrição for FAILED.

Observe que esse evento se aplica somente à operação de API [StartTranscriptionJob](#).

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Transcribe Job State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "TranscriptionJobName": "my-first-transcription-job",
    "TranscriptionJobStatus": "COMPLETED" (or "FAILED")
  }
}
```

- `TranscriptionJobName`: o nome exclusivo que você escolheu para o trabalho de transcrição.
- `TranscriptionJobStatus`: o status do trabalho de transcrição. Pode ser COMPLETED ou FAILED.

Eventos de identificação de idiomas

Quando você habilita a [identificação automática de idioma](#), o Amazon Transcribe gera um evento quando o estado de identificação do idioma é COMPLETED ou FAILED. Para identificar o trabalho que mudou de estado e iniciar o evento no destino, use o campo `JobName` do evento. Um evento do Amazon Transcribe contém as seguintes informações. Um campo `FailureReason` será adicionado em `detail` se o status de identificação do idioma for FAILED.

Observe que esse evento se aplica somente à operação de API [StartTranscriptionJob](#) quando o parâmetro [LanguageIdSettings](#) é incluído.

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Language Identification State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "JobType": "TranscriptionJob",
    "JobName": "my-first-lang-id-job",
    "LanguageIdentificationStatus": "COMPLETED" (or "FAILED")
  }
}
```

- JobType: para trabalhos de transcrição, esse valor deve ser TranscriptionJob.
- JobName: o nome exclusivo do trabalho de transcrição.
- LanguageIdentificationStatus: o status da identificação de idioma em um trabalho de transcrição. Pode ser COMPLETED ou FAILED.

Eventos do Call Analytics

Quando o estado de um trabalho do [Call Analytics](#) muda de IN_PROGRESS para COMPLETED ou FAILED, o Amazon Transcribe gera um evento. Para identificar o trabalho do Call Analytics que mudou de estado e iniciou o evento no destino, use o campo JobName do evento. Um evento do Amazon Transcribe contém as seguintes informações. Um campo FailureReason será adicionado em detail se o status do trabalho do Call Analytics for FAILED.

Observe que esse evento se aplica somente à operação de API [StartCallAnalyticsJob](#).

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Call Analytics Job State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
```

```

    "detail": {
      "JobName": "my-first-analytics-job",
      "JobStatus": "COMPLETED" (or "FAILED"),
      "AnalyticsJobDetails": { // only when you enable optional features such as
Generative Call Summarization
        "Skipped": []
      }
    }
  }
}

```

- **JobName**: o nome exclusivo do trabalho de transcrição do Call Analytics.
- **JobStatus**: o status do trabalho de transcrição do Call Analytics. Ele pode ser COMPLETED ou FAILED.
- **AnalyticsJobDetails**: os detalhes do seu trabalho de transcrição do Call Analytics, incluindo informações sobre os recursos de análise ignorados.

Eventos pós-chamada do Call Analytics

Quando uma transcrição da [análise pós-chamada](#) muda do estado IN_PROGRESS para COMPLETED ou FAILED, o Amazon Transcribe gera um evento. Para identificar o trabalho pós-chamada do Call Analytics que mudou de estado e iniciar o evento no destino, use o campo `StreamingSessionId` do evento.

Observe que esse evento se aplica somente à operação de API [StartCallAnalyticsStreamTranscription](#) quando o parâmetro [PostCallAnalyticsSettings](#) é incluído.

Um evento COMPLETED contém as seguintes informações:

```

{
  "version": "0",
  "id": "event ID",
  "detail-type": "Call Analytics Post Call Job State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "StreamingSessionId": "session-id",

```

```

    "PostCallStatus": "COMPLETED",
    "Transcript": {
      "RedactedTranscriptFileUri": "s3://DOC-EXAMPLE-BUCKET/my-output-files/my-
redacted-file.JSON",
      "TranscriptFileUri": "s3://DOC-EXAMPLE-BUCKET/my-output-files/my-file.JSON"
    },
    "Media": {
      "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-output-files/my-redacted-
file.WAV",
      "RedactedMediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-output-files/my-
redacted-file.WAV"
    }
  }
}

```

Um evento FAILED contém as seguintes informações:

```

{
  "version": "0",
  "id": "event ID",
  "detail-type": "Call Analytics Post Call Job State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "StreamingSessionId": "session-id",
    "PostCallStatus": "FAILED"
  }
}

```

- **StreamingSessionId**: o número de identificação atribuído à solicitação de transcrição do Call Analytics em tempo real.
- **PostCallStatus**: o status da transcrição do Call Analytics de pós-chamada. Ele pode ser COMPLETED ou FAILED.
- **Transcript**: o URI das transcrições editadas e não editadas.
- **Media**: o URI dos arquivos de áudio editados e não editados.

Eventos de vocabulário

Quando o estado de um [vocabulário personalizado](#) muda de PENDING para READY ou FAILED, Amazon Transcribe gera um evento. Para identificar o vocabulário personalizado que mudou de estado e iniciar o evento no destino, use o campo VocabularyName do evento. Um Amazon Transcribe evento contém as seguintes informações. Um campo FailureReason será adicionado em detail se o estado do vocabulário personalizado for FAILED.

Observe que esse evento se aplica somente à operação de API [CreateVocabulary](#).

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "Vocabulary State Change",
  "source": "aws.transcribe",
  "account": "111122223333",
  "time": "timestamp",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "VocabularyName": "unique-vocabulary-name",
    "VocabularyState": "READY" (or "FAILED")
  }
}
```

- VocabularyName: o nome exclusivo do vocabulário personalizado.
- VocabularyState: o estado de processamento do vocabulário personalizado. Pode ser READY ou FAILED.


Validação de conformidade para Amazon Transcribe

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte [Serviços da AWS Escopo por Programa de Conformidade](#) [Serviços da AWS](#) e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de [AWS conformidade Programas AWS](#) de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte [Baixar relatórios em AWS Artifact](#) .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. AWS fornece os seguintes recursos para ajudar na conformidade:

- [Guias de início rápido sobre segurança e conformidade](#) — Esses guias de implantação discutem considerações arquitetônicas e fornecem etapas para a implantação de ambientes básicos AWS focados em segurança e conformidade.
- [Arquitetura para segurança e conformidade com a HIPAA na Amazon Web Services](#) — Este whitepaper descreve como as empresas podem usar AWS para criar aplicativos qualificados para a HIPAA.

 Note

Nem todos Serviços da AWS são elegíveis para a HIPAA. Para obter mais informações, consulte a [Referência dos serviços qualificados pela HIPAA](#).

- AWS Recursos de <https://aws.amazon.com/compliance/resources/> de conformidade — Essa coleção de pastas de trabalho e guias pode ser aplicada ao seu setor e local.
- [AWS Guias de conformidade do cliente](#) — Entenda o modelo de responsabilidade compartilhada sob a ótica da conformidade. Os guias resumem as melhores práticas de proteção Serviços da AWS e mapeiam as diretrizes para controles de segurança em várias estruturas (incluindo o Instituto Nacional de Padrões e Tecnologia (NIST), o Conselho de Padrões de Segurança do Setor de Cartões de Pagamento (PCI) e a Organização Internacional de Padronização (ISO)).
- [Avaliação de recursos com regras](#) no Guia do AWS Config desenvolvedor — O AWS Config serviço avalia o quão bem suas configurações de recursos estão em conformidade com as práticas internas, as diretrizes e os regulamentos do setor.
- [AWS Security Hub](#) — Isso AWS service (Serviço da AWS) fornece uma visão abrangente do seu estado de segurança interno AWS. O Security Hub usa controles de segurança para avaliar os atributos da AWS e verificar a conformidade com os padrões e as práticas recomendadas do setor de segurança. Para obter uma lista dos serviços e controles aceitos, consulte a [Referência de controles do Security Hub](#).
- [Amazon GuardDuty](#) — Isso AWS service (Serviço da AWS) detecta possíveis ameaças às suas cargas de trabalho Contas da AWS, contêineres e dados monitorando seu ambiente em busca de atividades suspeitas e maliciosas. GuardDuty pode ajudá-lo a atender a vários requisitos de conformidade, como o PCI DSS, atendendo aos requisitos de detecção de intrusões exigidos por determinadas estruturas de conformidade.

- [AWS Audit Manager](#)— Isso AWS service (Serviço da AWS) ajuda você a auditar continuamente seu AWS uso para simplificar a forma como você gerencia o risco e a conformidade com as regulamentações e os padrões do setor.

Resiliência no Amazon Transcribe

A infraestrutura global da AWS se baseia em Regiões da AWS e zonas de disponibilidade. A Regiões da AWS oferece várias zonas de disponibilidade separadas e isoladas fisicamente que são conectadas com baixa latência, throughputs elevadas e em redes altamente redundantes. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre Regiões da AWS e zonas de disponibilidade, consulte [Infraestrutura global da AWS](#).

Segurança da infraestrutura no Amazon Transcribe

Como um serviço gerenciado, Amazon Transcribe é protegido pela segurança de rede AWS global. Para obter informações sobre serviços de segurança da AWS e como a AWS protege a infraestrutura, consulte [Segurança na Nuvem AWS](#). Para projetar seu ambiente da AWS usando as práticas recomendadas de segurança de infraestrutura, consulte [Proteção de infraestrutura](#) em Pilar segurança: AWS Well-Architected Framework.

Você usa chamadas de API publicadas pela AWS para acessar o Amazon Transcribe por meio da rede. Os clientes devem oferecer suporte para:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Além disso, as solicitações devem ser assinadas usando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou você pode usar o [AWS Security Token Service](#) (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Análise e gerenciamento de vulnerabilidades no Amazon Transcribe

A configuração e os controles de TI são uma responsabilidade compartilhada entre a AWS e você, nosso cliente. Para obter mais informações, consulte o [modelo de responsabilidade compartilhada da AWS](#).

Amazon Transcribe e VPC endpoints de interface (AWS PrivateLink)

É possível estabelecer uma conexão privada entre a VPC e o Amazon Transcribe criando um VPC endpoint de interface. Os endpoints de interface são alimentados por [AWS PrivateLink](#), uma tecnologia que você pode usar para acessar de forma privada Amazon Transcribe APIs sem gateway de internet, dispositivo NAT, conexão VPN ou AWS Direct Connect conexão. As instâncias na VPC não precisam de endereços IP públicos para a comunicação com APIs do Amazon Transcribe. O tráfego de rede entre a VPC e o Amazon Transcribe não deixa a rede da Amazon.

Cada endpoint de interface é representado por uma ou mais [interfaces de rede elástica](#) nas sub-redes.

Para obter mais informações, consulte [Endpoints da interface da VPC \(AWS PrivateLink\)](#) no Manual do usuário do Amazon VPC.

Considerações sobre endpoints da VPC do Amazon Transcribe

Antes de configurar um endpoint VPC de interface para Amazon Transcribe, certifique-se de revisar [Propriedades e limitações do endpoint da interface](#) na Amazon VPC Guia do usuário.

O Amazon Transcribe oferece suporte a chamadas para todas as ações de API da VPC.

Criar um VPC endpoint de interface para o Amazon Transcribe

Você pode criar um endpoint VPC para o Amazon Transcribe serviço usando o Amazon VPC AWS Management Console ou AWS CLI. Para obter mais informações, consulte [Criar um endpoint de interface](#) no Guia do usuário da Amazon VPC.

Para transcrições em lote em Amazon Transcribe, crie um VPC endpoint usando o seguinte nome de serviço:

- `com.amazonaws.us-west-2.transcrever`

Para streaming de transcrições em Amazon Transcribe, crie um VPC endpoint usando o seguinte nome de serviço:

- `com.amazonaws.us-west-2.transcribe-streaming`

Se você habilitar o DNS privado para o endpoint, poderá fazer solicitações de API para Amazon Transcribe usando seu nome DNS padrão para a Região da AWS, por exemplo, `transcribestreaming.us-east-2.amazonaws.com`.

Para obter mais informações, consulte [Acessando um serviço por meio de um endpoint de interface na Amazon VPC](#) Guia do usuário.

Criar uma política de VPC endpoint para o Amazon Transcribe

Você pode anexar uma política de endpoint ao seu endpoint VPC que controla o acesso ao serviço de streaming ou ao serviço de transcrição em lote do Amazon Transcribe. Essa política especifica as seguintes informações:

- A entidade principal que pode executar ações.
- As ações que podem ser executadas.
- Os recursos sobre os quais as ações podem ser realizadas.

Para obter mais informações, consulte [Controlar o acesso a serviços com VPC endpoints](#) no Guia do usuário da Amazon VPC.

Exemplo: política de endpoint de VPC para Amazon Transcribe ações de transcrição em lote

Veja a seguir um exemplo de uma política de endpoint para uma transcrição em lote no Amazon Transcribe. Quando anexada a um endpoint, essa política concede acesso às ações indicadas do Amazon Transcribe para todos os principais em todos os recursos.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "transcribe:StartTranscriptionJob",
        "transcribe:ListTranscriptionJobs"
      ]
    }
  ]
}
```

```

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

```

Exemplo: política de endpoint de VPC para ações de transcrição de streaming

Veja a seguir um exemplo de uma política de endpoint para uma transcrição de streaming no Amazon Transcribe. Quando anexada a um endpoint, essa política concede acesso às ações indicadas do Amazon Transcribe para todos os principais em todos os recursos.

```

{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "transcribe:StartStreamTranscription",
        "transcribe:StartStreamTranscriptionWebsocket"
      ],
      "Resource": "*"
    }
  ]
}

```

Sub-redes compartilhadas

Você não pode criar, descrever, modificar ou excluir endpoints VPC em sub-redes que são compartilhadas com você. No entanto, você pode usar os endpoints da VPC em sub-redes que são compartilhadas com você. Para obter informações sobre o compartilhamento de VPC, consulte [Compartilhe sua VPC com outras contas](#) na Amazon Virtual Private Cloud guia.

Melhores práticas de segurança do Amazon Transcribe

As práticas recomendadas a seguir são diretrizes gerais e não representam uma solução completa de segurança. Como essas práticas recomendadas podem não ser adequadas ou suficientes no seu ambiente, use-as como considerações úteis em vez de requisitos.

- Use criptografia de dados, como contexto AWS KMS de criptografia

AWS KMS. O contexto de criptografia é um mapa de pares chave:valor não secretos e em texto simples. Esse mapa representa dados autenticados adicionais, conhecidos como pares de contexto de criptografia, que fornecem uma camada adicional de segurança para seus dados.

Para obter mais informações, consulte [Contexto de criptografia do AWS KMS](#).

- Use credenciais temporárias sempre que possível

Sempre que possível, use credenciais temporárias em vez de credenciais de longo prazo, como chaves de acesso. Para cenários em que você precise de IAM usuários com acesso programático e credenciais de longo prazo, recomendamos alternar as chaves de acesso. A modificação regular de credenciais de longo prazo ajuda você a se familiarizar com o processo. Isso é útil caso você esteja em uma situação em que precise alternar credenciais, como quando um funcionário deixa sua empresa. Recomendamos que você use as últimas informações de IAM acesso para alternar e remover chaves de acesso com segurança.

Para obter mais informações, consulte [Chaves de acesso rotativas](#) e [melhores práticas de segurança em IAM](#).

- Use IAM funções para aplicativos e AWS serviços que exigem Amazon Transcribe acesso

Use uma IAM função para gerenciar credenciais temporárias para aplicativos ou serviços que precisam de acesso ao Amazon Transcribe. Quando você usa uma função, não precisa distribuir credenciais de longo prazo, como chaves de acesso ou chaves de acesso, para uma Amazon EC2 instância do ou AWS serviço da como o. IAM as funções podem fornecer permissões temporárias que os aplicativos podem usar ao fazer solicitações de AWS recursos da.

Para obter mais informações, consulte [IAM Rastreio de acesso a recursos de IAM e serviços](#).

- Use o controle de acesso com base em tags

Você pode usar etiquetas para controlar o acesso ao Contas da AWS. As tags In Amazon Transcribe. podem ser adicionadas a: trabalhos de transcrição, vocabulários personalizados, filtros de vocabulário personalizados e modelos de linguagem personalizados.

Para obter mais informações, consulte [Controle de acesso com base em tags](#).

- Use ferramentas AWS de monitoramento

O monitoramento é uma parte importante da manutenção da confiabilidade, da segurança, da disponibilidade e da performance do Amazon Transcribe e das suas soluções da AWS. Você pode monitorar Amazon Transcribe usando CloudTrail.

Para obter mais informações, consulte [Monitoramento Amazon Transcribe com AWS CloudTrail](#).

- Habilitar AWS Config

AWS Config pode avaliar, auditar e verificar as configurações dos recursos da AWS. Usando AWS Config, você pode revisar as alterações nas configurações e nos relacionamentos entre os recursos. Você também pode examinar os detalhes do histórico de configuração de recursos e determinar a conformidade geral em relação às configurações especificadas em diretrizes internas. Isso pode ajudar a simplificar a auditoria de conformidade, a análise de segurança, o gerenciamento de alterações e a solução de problemas operacionais.

Para obter mais informações, consulte [O que é AWS Config?](#)

Amazon Transcribe Medical

Amazon Transcribe O Medical é um serviço de reconhecimento automático de fala (ASR) desenvolvido para profissionais médicos que desejam transcrever discursos médicos, como anotações ditadas por médicos, monitoramento da segurança de medicamentos, consultas de telemedicina ou conversas médico-paciente. Amazon Transcribe O Medical está disponível por meio de streaming em tempo real (via microfone) ou transcrição de um arquivo carregado (lote).

Important

Amazon Transcribe A medicina não substitui o aconselhamento, diagnóstico ou tratamento médico profissional. Identifique o limite de confiança certo para seu caso de uso e use limites de alta confiança em situações que exigem alta precisão. Para certos casos de uso, os resultados devem ser revisados e verificados por revisores humanos devidamente treinados. Amazon Transcribe As transcrições médicas só devem ser usadas em cenários de atendimento ao paciente após análise para precisão e bom julgamento médico por profissionais médicos treinados.

Amazon Transcribe A Medical opera sob um modelo de responsabilidade compartilhada, em que AWS é responsável por proteger a infraestrutura que administra o Amazon Transcribe Medical e você é responsável por gerenciar seus dados. Para obter mais informações, consulte o [Modelo de responsabilidade compartilhada](#).

Amazon Transcribe Medical está disponível em inglês americano (en-US).

Para obter melhores resultados, use um formato de áudio sem perdas, como FLAC ou WAV, com codificação PCM de 16 bits. Amazon Transcribe Medical suporta taxas de amostragem de 16.000 Hz ou mais.

Para análise de suas transcrições, você pode usar outros Serviços da AWS, como [Amazon Comprehend Medical](#).

Especialidades suportadas

Especialidade	Subespecialidade	Entrada de áudio
Cardiologia	none	somente streaming

Especialidade	Subespecialidade	Entrada de áudio
Neurologia	none	somente streaming
Oncologia	none	somente streaming
Cuidados Principal	Medicina da família	em lote de de lote de.
Cuidados Principal	Medicina interna	em lote de de lote de.
Cuidados Principal	Obstetrícia e ginecologia (OB-GYN)	em lote de de lote de.
Cuidados Principal	Pediatria	em lote de de lote de.
Radiologia	none	somente streaming
Urologia	none	somente streaming

Disponibilidade e cotas da região

A análise de chamadas é compatível com o seguinte Regiões da AWS:

Região	Tipo de transcrição
af-south-1: Cidade do Cabo	lote
ap-east-1 (Hong Kong)	lote
ap-northeast-1 (Tóquio)	em lote de de lote de.
ap-northeast-2 (Seul)	em lote de de lote de.
ap-south-1: Mumbai)	lote
ap-southeast-1 (Cingapura)	lote
ap-southeast-2 (Sydney)	em lote de de lote de.
ca-central-1: Canadá, Central)	em lote de de lote de.

Região	Tipo de transcrição
eu-central-1 (Frankfurt)	em lote de de lote de.
eu-north-1 (Estocolmo)	lote
eu-west-1 (Irlanda)	em lote de de lote de.
eu-west-2 (Londres)	em lote de de lote de.
eu-west-3 (Paris)	lote
me-south-1: Bahrein)	lote
sa-east-1 (São Paulo)	em lote de de lote de.
us-east-1 (Norte da Virgínia)	em lote de de lote de.
us-east-2 (Ohio)	em lote de de lote de.
us-gov-east-1 (GovCloud, Leste dos EUA)	em lote de de lote de.
us-gov-west-1 (GovCloud, Oeste dos EUA)	em lote de de lote de.
us-west-1 (São Francisco)	lote
us-west-2 (Oregon)	em lote de de lote de.

Observe que o suporte regional difere para [Amazon Transcribe](#) Amazon Transcribe Medical, e [Call Analytics](#).

Para obter os endpoints de cada região compatível, consulte [Endpoints de serviço](#) na ReferênciaAWS Geral.

Para obter uma lista de cotas relacionadas às suas transcrições, consulte as [cotas de serviço](#) na ReferênciaAWS Geral. Algumas cotas podem ser alteradas mediante solicitação. Se a coluna Ajustável contiver “Sim”, você poderá solicitar um aumento. Para fazer isso, selecione o link fornecido.

Conceitos e termos termos

Ao criar um trabalho de transcrição médica, especifique o idioma, a especialidade médica e o tipo de áudio do arquivo de origem. Insira inglês dos EUA (en-US) como o idioma e PRIMARYCARE como a especialidade médica. Inserir a atenção primária como o valor permite gerar transcrições do áudio de origem nas seguintes especialidades médicas:

- Medicina da família
- Medicina interna
- Obstetrícia e ginecologia (OB-GYN)
- Pediatria

Você pode escolher entre ditado e conversa para o tipo de áudio. Escolha ditado para arquivos de áudio onde o médico está passando um relatório sobre uma consulta ou um procedimento do paciente. Escolha conversa para arquivos de áudio que envolvam uma conversa entre um médico e um paciente ou entre médicos.

Para armazenar o resultado do trabalho de transcrição, selecione um bucket do Amazon S3 que você já criou. Para obter mais informações sobre Amazon S3 buckets, consulte [Getting Started with Amazon Simple Storage Service](#).

Veja a seguir o número mínimo de parâmetros de solicitação a serem inseridos no exemplo de JSON.

```
{
  "MedicalTranscriptionJobName": "my-first-transcription-job",
  "LanguageCode": "en-US",
  "Media": {
    "MediaFileUri": "s3://path to your audio file"
  },
  "OutputBucketName": "your output bucket name",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION"
}
```

Amazon TranscribeO Medical permite que você gere transcrições alternativas. Para obter mais informações, consulte [Gerando transcrições alternativas](#).

Você também pode ativar o particionamento de alto-falantes ou identificar canais em seu áudio. Para obter mais informações, consulte [Habilitar particionamento de alto-falantes](#) e [Transcrevendo áudio multicanal](#).

Transcrevendo termos e medidas médicas

Amazon TranscribeO médico pode transcrever termos e medidas médicas. Amazon Transcribe Abreviações de saídas médicas para termos falados. Por exemplo, "pressão arterial" é transcrita como PA. Você pode encontrar uma lista de convenções que aAmazon Transcribe Medicina usa para termos e medidas médicas na tabela desta página. A coluna Termo falado refere-se ao termo falado no áudio de origem. A coluna Resultado refere-se à abreviatura que você verá nos resultados da transcrição.

Você pode ver como os termos falados no áudio fonte correspondem ao resultado da transcrição aqui.

Termo falado no áudio de origem	Abreviatura usada na saída	Exemplo de saída
Centígrado	C	A temperatura do paciente é de 37,4 C.
Celsius	C	A temperatura do paciente é de 37,4 C.
Fahrenheit	F	A temperatura do paciente é 101 F.
gramas	g	Uma massa de 100 g foi extraída do paciente.
metros	m	O paciente tem 1,8 m de altura.
pés	ft	O paciente tem 6 ft de altura.
Quilos	kg	O paciente pesa 80 kg.
quilogramas	kg	O paciente pesa 80 kg.

Termo falado no áudio de origem	Abreviatura usada na saída	Exemplo de saída
c c	cc	O paciente recebeu 100 cc de solução salina.
centímetro cúbico	cc	O paciente recebeu 100 cc de solução salina.
mililitro	mL	O paciente excretou 100 mL de urina.
pressão arterial	PA	A pressão arterial do paciente estava alta.
p a	PA	A pressão arterial do paciente estava alta.
X por Y	X/Y	A PA do paciente era de 120/80.
batimentos por min	BPM	O paciente teve fibrilação atrial com frequência cardíaca de 160 BPM.
batimentos por minuto	BPM	O paciente teve fibrilação atrial com frequência cardíaca de 160 BPM.
O 2	O2	A saturação de O2 do paciente estava em 98%.
CO2	CO2	O paciente precisou de suporte respiratório para o aumento de CO2.
pós-operação	POSTOP	O paciente veio para avaliação de POSTOP.

Termo falado no áudio de origem	Abreviatura usada na saída	Exemplo de saída
pós-operatório	POSTOP	O paciente veio para avaliação de POSTOP.
tomografia computadorizada	Tomografia computadorizada	Como o paciente parecia ter sofrido uma hemorragia cerebral, foi necessário fazer uma tomografia computadorizada.
Pulso 80	P 80	Os sinais vitais do paciente foram P 80, R 17,...
Respiração 17	R 17	Os sinais vitais do paciente foram P 80, R 17,...
arritmia	E/S	O paciente apresentou arritmia sinusal
L cinco	L5	A punção lombar foi realizada entre a L4 e a L5

Transcrevendo números

Amazon TranscribeO médico transcreve dígitos como números em vez de palavras. Por exemplo, o número falado "mil duzentos e quarenta e dois" é transcrito como 1242.

Os números são transcritos de acordo com as seguintes regras.

Rule	Descrição
Converter números cardinais maiores que 10 em números.	<ul style="list-style-type: none"> "Cinquenta e cinco" > 55 "cem" > 100 "Mil e trinta um" > 1031

Rule	Descrição
	<ul style="list-style-type: none"> • “Cento e vinte e três milhões, quatrocentos e cinquenta e seis mil, setecentos e oitenta e nove” > 123.456.789
<p>Converter números cardinais seguidos de "milhão" ou "bilhão" em numerais seguidos de uma palavra quando "milhão" ou "bilhão" não for seguido por um número.</p>	<ul style="list-style-type: none"> • “cem milhões” > 100 milhões • “um bilhão” > 1 bilhão • “dois vírgula três milhões” > 2,3 milhões
<p>Converter números ordinais maiores que 10 em números.</p>	<ul style="list-style-type: none"> • "Quadragésimo terceiro" > 43º • "vigésima sexta avenida" > 26ª avenida
<p>Converter frações no respectivo formato numérico.</p>	<ul style="list-style-type: none"> • "um quarto" > 1/4 • "três dezesseis avos" > 3/16 • "um meio" > 1/2 • "um centésimo" > 1/100
<p>Converter números menores que 10 dígitos se houver mais de um em uma linha.</p>	<ul style="list-style-type: none"> • "três quatro cinco" > 345 • "Meu número de telefone é quatro dois cinco cinco cinco cinco um dois um dois" > 4255551212
<p>Os decimais são indicados por "ponto".</p>	<ul style="list-style-type: none"> • "trezentos e três ponto cinco" > 303.5 • "três ponto vinte e três" > 3.23 • "zero ponto quatro" > 0.4 • "zero ponto três" > 0.3
<p>Converter a palavra "porcento" após um número em um sinal de porcentagem (%).</p>	<ul style="list-style-type: none"> • "vinte e três por cento" > 23% • "vinte e três ponto quatro cinco por cento" > 23.45%
<p>Converter as palavras "dólar", "dólar americano", "dólar australiano", "AUD" ou "USD" após um número em um cifrão (\$) antes do número.</p>	<ul style="list-style-type: none"> • "um dólar e quinze centavos" > \$ 1,15 • “vinte e três USD” > \$ 23 • “vinte e três dólares australianos” > \$ 23

Rule	Descrição
Converter as palavras "libras" ou "miligramas" em "lbs" ou "mg",	<ul style="list-style-type: none"> "vinte e três libras" > 23 lbs "quarenta e cinco miligramas" > 45 mg
Converter as palavras "rúpias", "rúpias indianas" ou "INR" após um número no símbolo de rúpia (#) antes do número.	<ul style="list-style-type: none"> "vinte e três rúpias" > #23 "cinquenta rúpias e trinta paisas" > # 50,30
Converter horas em números.	<ul style="list-style-type: none"> "sete a m do horário padrão do leste" > 7 am do horário padrão do leste "doze e trinta p m" > 12:30 p.m:
<p>Combinar anos expressos como dois dígitos em quatro.</p> <p>Válido somente para os séculos 20, 21 e 22.</p>	<ul style="list-style-type: none"> "mil novecentos e sessenta e dois" > 1962 "o ano é dois mil e doze" > o ano é 2012 "dois mil e dezenove" > 2019 "dois mil cento e trinta" > 2130
Converter datas em números.	<ul style="list-style-type: none"> "quinto dia de maio de dois mil e doze" > 5 de maio de 2012 "cinco de maio de dois mil e doze" > 5 de maio de 2012 "cinco maio dois mil e doze" > 5 de maio de 2012
Separar intervalos de números pela palavra "a".	<ul style="list-style-type: none"> "vinte e três a trinta sete" > 23 a 37

Transcrevendo uma conversa médica

Você pode usar oAmazon Transcribe Medical para transcrever uma conversa médica entre um médico e um paciente usando um trabalho de transcrição em lote ou um stream em tempo real. Os trabalhos de transcrição em Batch permitem que você transcreva arquivos de áudio. Para garantir que oAmazon Transcribe Medical produza resultados de transcrição com a maior precisão possível, você deve especificar a especialidade médica do médico em seu trabalho ou fluxo de transcrição.

Você pode transcrever uma visita médico-paciente nas seguintes especialidades médicas:

- Cardiologia — disponível somente na transcrição em streaming
- Neurologia — disponível somente na transcrição de streaming
- Oncologia — disponível somente na transcrição em streaming
- Atenção primária — inclui os seguintes tipos de prática médica:
 - Medicina da família
 - Medicina interna
 - Obstetrícia e ginecologia (OB-GYN)
 - Pediatria
- Urologia — disponível somente na transcrição em streaming

Você pode melhorar a precisão da transcrição usando vocabulários médicos personalizados. Para obter informações sobre como funcionam os vocabulários médicos personalizados, consulte [Melhorando a precisão da transcrição com vocabulários médicos personalizados](#).

Por padrão, o Amazon Transcribe Medical retorna a transcrição com o nível de confiança mais alto. Se você quiser configurá-lo para retornar transcrições alternativas, consulte [Gerando transcrições alternativas](#).

Para obter informações sobre como os números e as medidas médicas aparecem na saída da transcrição, consulte [Transcrevendo números](#) e [Transcrevendo termos e medidas médicas](#).

Tópicos

- [Transcrevendo um arquivo de áudio de uma conversa médica](#)
- [Transcrevendo uma conversa médica em um stream em tempo real](#)
- [Habilitar particionamento de alto-falantes](#)
- [Transcrevendo áudio multicanal](#)

Transcrevendo um arquivo de áudio de uma conversa médica

Use um trabalho de transcrição em lote para transcrever arquivos de áudio de conversas médicas. Você pode usar isso para transcrever um diálogo médico-paciente. Você pode iniciar um trabalho de transcrição em lote na [StartMedicalTranscriptionJob](#) API ou no AWS Management Console.

Ao iniciar um trabalho de transcrição médica com a [StartMedicalTranscriptionJob](#) API, você especifica `PRIMARYCARE` como o valor do `Specialty` parâmetro.

AWS Management Console

Para transcrever um diálogo médico-paciente (AWS Management Console)

Para usar o diálogoAWS Management Console para transcrever um diálogo médico-paciente, crie um trabalho de transcrição e escolha Conversa para o tipo de entrada de áudio.

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, emAmazon Transcribe Principal, escolha Trabalhos de transcrição.
3. Escolha Create job (Criar trabalho).
4. Na página Especificar detalhes do Job, em Configurações do trabalho, especifique o seguinte.
 - a. Nome — o nome do trabalho de transcrição.
 - b. Tipo de entrada de áudio — Conversa
5. Para os campos restantes, especifique aAmazon S3 localização do seu arquivo de áudio e onde você deseja armazenar a saída do trabalho de transcrição.
6. Escolha Next (Próximo).
7. Escolha Create (Criar).

API

Para transcrever uma conversa médica usando um trabalho de transcrição em lote (API)

- Para a [StartMedicalTranscriptionJob](#)API, especifique o seguinte.
 - a. Para`MedicalTranscriptionJobName`, especifique um nome exclusivo em seuConta da AWS.
 - b. Para`LanguageCode`, especifique o código do idioma que corresponde ao idioma falado no seu arquivo de áudio e ao idioma do seu filtro de vocabulário.
 - c. Para o`MediaFileUri` parâmetro do`Media` objeto, especifique o nome do arquivo de áudio que você deseja transcrever.
 - d. Para`Specialty`, especifique a especialidade médica do médico que fala no arquivo de áudio como`PRIMARYCARE`.
 - e. Para `Type`, especifique `CONVERSATION`.
 - f. Para`OutputBucketName`, especifique oAmazon S3 bucket para armazenar os resultados da transcrição.

A seguir está um exemplo de solicitação que usa oAWS SDK for Python (Boto3) para transcrever uma conversa médica de um médico daPRIMARYCARE especialidade e de um paciente.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-med-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION'
)

while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

O código de exemplo a seguir mostra os resultados da transcrição de uma conversa médico-paciente.

```
{
  "jobName": "conversation-medical-transcription-job",
```

```
"accountId": "111122223333",
"results": {
  "transcripts": [
    {
      "transcript": "... come for a follow up visit today..."
    }
  ],
  "items": [
    {
      ...
      "start_time": "4.85",
      "end_time": "5.12",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "come"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "5.12",
      "end_time": "5.29",
      "alternatives": [
        {
          "confidence": "1.0",
          "content": "for"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "5.29",
      "end_time": "5.33",
      "alternatives": [
        {
          "confidence": "0.9955",
          "content": "a"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "5.33",
```

```
        "end_time": "5.66",
        "alternatives": [
            {
                "confidence": "0.9754",
                "content": "follow"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.66",
        "end_time": "5.75",
        "alternatives": [
            {
                "confidence": "0.9754",
                "content": "up"
            }
        ],
        "type": "pronunciation"
    },
    {
        "start_time": "5.75",
        "end_time": "6.02",
        "alternatives": [
            {
                "confidence": "1.0",
                "content": "visit"
            }
        ]
    }
    ...
},
"status": "COMPLETED"
}
```

AWS CLI

Para transcrever uma conversa médica usando um trabalho de transcrição em lote (AWS CLI)

- Execute o código a seguir.

```
aws transcribe start-medical-transcription-job \  
--region us-west-2 \  
--cli-input-json file://example-start-command.json
```

O código a seguir mostra o conteúdo de `example-start-command.json`.

```
{  
  "MedicalTranscriptionJobName": "my-first-med-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION"  
}
```

Transcrevendo uma conversa médica em um stream em tempo real

Você pode transcrever um fluxo de áudio de uma conversa médica usando o HTTP/2 ou [WebSocket](#) protocolos. Para obter informações sobre como iniciar um stream usando o WebSocket protocolo, consulte [Configurando um WebSocket stream](#). Para iniciar um stream HTTP/2, use a [StartMedicalStreamTranscriptionAPI](#).

Você pode transcrever streaming de áudio nas seguintes especialidades médicas:

- Cardiologia
- Neurologia
- Oncologia
- Principal de atenção
- Urologia

Cada especialidade médica inclui muitos tipos de procedimentos e consultas. Os médicos, portanto, ditam muitos tipos diferentes de notas. Use os exemplos a seguir como orientação para ajudá-

lo a especificar o valor do parâmetro `specialty` URI da WebSocket solicitação ou o `specialty` parâmetro da [StartMedicalStreamTranscription](#) API:

- Para consultas de eletrofisiologia ou ecocardiografia, escolha `CARDIOLOGY`.
- Para consultas de oncologia médica, oncologia cirúrgica ou oncologia por radiação, escolha `ONCOLOGY`.
- Para um médico que presta uma consulta a um paciente que teve um derrame, seja um ataque isquêmico transitório ou um ataque cerebrovascular, escolha `NEUROLOGY`.
- Para uma consulta sobre incontinência urinária, escolha `UROLOGY`.
- Para exames anuais ou consultas de atendimento urgente, escolha `PRIMARYCARE`.
- Para visitas hospitalares internadas, escolha `PRIMARYCARE`.
- Para consultas sobre fertilidade, laqueadura tubária, inserção do DIU ou aborto, escolha `PRIMARYCARE`.

AWS Management Console

Para transcrever uma conversa médica em streaming (AWS Management Console)

Para usar o AWS Management Console para transcrever um diálogo médico-paciente em tempo real, escolha a opção de transcrever uma conversa médica, iniciar a transmissão e começar a falar no microfone.

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, em Amazon Transcribe Principal, escolha Transcrição em tempo real.
3. Escolha Conversa.
4. Para especialidade médica, escolha a especialidade do médico.
5. Selecione Start streaming (Iniciar transmissão).
6. Fale no microfone.

Transcrevendo uma conversa médica em um stream HTTP/2

Veja a seguir a sintaxe dos parâmetros de uma solicitação HTTP/2.

Para transcrever um fluxo HTTP/2 de uma conversa médica, use a [StartMedicalStreamTranscription](#) API e especifique o seguinte:

- `LanguageCode`— O código do idioma. O valor válido é `en-US`.
- `MediaEncoding`— A codificação usada para o áudio de entrada. Os valores válidos são `pcm`, `ogg-opus` e `flac`.
- `Specialty`— A especialidade do profissional médico.
- `Type` – `CONVERSATION`

Para melhorar a precisão da transcrição de termos específicos em um stream em tempo real, use um vocabulário personalizado. Para ativar um vocabulário personalizado, defina o valor do `VocabularyName` parâmetro como o nome do vocabulário personalizado que você deseja usar. Para obter mais informações, consulte [Melhorando a precisão da transcrição com vocabulários médicos personalizados](#).

Para rotular a fala de diferentes alto-falantes, defina o `ShowSpeakerLabel` parâmetro como `true`. Para obter mais informações, consulte [Habilitar particionamento de alto-falantes](#).

Para obter mais informações sobre como configurar um stream HTTP/2 para transcrever uma conversa médica, consulte [Configurar um fluxo de HTTP/2](#).

Transcrevendo uma conversa médica em um WebSocket stream

Você pode usar uma WebSocket solicitação para transcrever uma conversa médica. Ao fazer uma WebSocket solicitação, você cria um URI pré-assinado. Esse URI contém as informações necessárias para configurar o fluxo de áudio entre seu aplicativo e o Amazon Transcribe Medical. Para obter mais informações sobre a criação WebSocket de solicitações, consulte [Configurando um WebSocket stream](#).

O modelo a seguir pode ser usado para criar o URI pré-assinado.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-  
transcription-websocket  
?language-code=LanguageCode  
&X-Amz-Algorithm=AWS4-HMAC-SHA256  
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-  
west-2%2Ftranscribe%2Faws4_request  
&X-Amz-Date=20220208T235959Z  
&X-Amz-Expires=300  
&X-Amz-Security-Token=security-token  
&X-Amz-Signature=Signature Version 4 signature  
&X-Amz-SignedHeaders=host  
&media-encoding=flac
```

```
&sample-rate=16000
&session-id=sessionId
&specialty=medicalSpecialty
&type=CONVERSATION
&vocabulary-name=vocabularyName
&show-speaker-label=boolean
```

Para melhorar a precisão da transcrição de termos específicos em um stream em tempo real, use um vocabulário personalizado. Para ativar um vocabulário personalizado, defina o valor de `vocabulary-name` como o nome do vocabulário personalizado que você deseja usar. Para obter mais informações, consulte [Melhorando a precisão da transcrição com vocabulários médicos personalizados](#).

Para rotular a fala de diferentes alto-falantes, defina o `show-speaker-label` parâmetro como `true`. Para obter mais informações, consulte [Habilitar particionamento de alto-falantes](#).

Para obter mais informações sobre a criação de URIs pré-assinados, consulte [Configurando um WebSocket stream](#).

Habilitar particionamento de alto-falantes

Para ativar o particionamento de alto-falantes em Amazon Transcribe Medicina, use a diarização de alto-falantes. Isso permite que você veja o que o paciente disse e o que o médico disse na saída da transcrição.

Quando você ativa a diarização do locutor, o Amazon Transcribe Medical rotula cada enunciado com um identificador exclusivo para cada locutor. Um enunciado é uma unidade de fala que normalmente é separada de outros enunciados pelo silêncio. Na transcrição em lote, um enunciado do médico poderia receber um rótulo `pk_0` e um enunciado do qual o paciente poderia receber um rótulo `pk_1`.

Se um enunciado de um orador se sobrepõe a um enunciado de outro orador, o Amazon Transcribe Medical o ordena na transcrição de acordo com o horário de início. As falas que se sobrepõem no áudio de entrada não se sobrepõem na saída da transcrição.

Você pode ativar a diarização do alto-falante ao transcrever um arquivo de áudio usando o trabalho de transcrição em lote ou em um stream em tempo real.

Tópicos

- [Habilitando o particionamento de alto-falantes em transcrições em lote](#)

- [Habilitando o particionamento de alto-falantes em streams em tempo real](#)

Habilitando o particionamento de alto-falantes em transcrições em lote

Você pode ativar o particionamento de alto-falantes em um trabalho de transcrição em lote usando a [StartMedicalTranscriptionJob](#) API ou o AWS Management Console. Isso permite que você particione o texto por palestrante em uma conversa médico-paciente e determine quem disse o quê na saída da transcrição.

AWS Management Console

Para usar o AWS Management Console para habilitar a diarização do alto-falante em seu trabalho de transcrição, você ativa a identificação de áudio e, em seguida, o particionamento do alto-falante.

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, em Amazon Transcribe Principal, escolha Trabalhos de transcrição.
3. Escolha Create job (Criar trabalho).
4. Na página Especificar detalhes do trabalho, forneça informações sobre seu trabalho de transcrição.
5. Escolha Próximo.
6. Ative a identificação de áudio.
7. Para Tipo de identificação de áudio, escolha Particionamento de alto-falante.
8. Em Número máximo de alto-falantes, insira o número máximo de alto-falantes que você acha que estão falando em seu arquivo de áudio.
9. Escolha Create (Criar).

API

Para habilitar o particionamento de alto-falantes usando um trabalho de transcrição em lote (API)

- Para a [StartMedicalTranscriptionJob](#) API, especifique o seguinte.
 - a. Para `MedicalTranscriptionJobName`, especifique um nome exclusivo em sua Conta da AWS.
 - b. Para `LanguageCode`, especifique o código do idioma que corresponde ao idioma falado no arquivo de áudio.

- c. Para `MediaFileUri` parâmetro do `Media` objeto, especifique o nome do arquivo de áudio que você deseja transcrever.
- d. Para `Specialty`, especifique a especialidade médica do médico que fala no arquivo de áudio.
- e. Para `Type`, especifique `CONVERSATION`.
- f. Para `OutputBucketName`, especifique o Amazon S3 bucket para armazenar os resultados da transcrição.
- g. Para o `Settings` objeto, especifique o seguinte.
 - i. `ShowSpeakerLabels` – `true`.
 - ii. `MaxSpeakerLabels`— Um número inteiro entre 2 e 10 para indicar o número de alto-falantes que você acha que estão falando em seu áudio.

A solicitação a seguir usa o AWS SDK for Python (Boto3) para iniciar um trabalho de transcrição em lote de um diálogo com o paciente de um médico de cuidados primários com a partição de alto-falantes ativada.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media={
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION',
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    Settings = {'ShowSpeakerLabels': True,
                'MaxSpeakerLabels': 2
               }
)
```

```
while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

O código de exemplo a seguir mostra os resultados da transcrição de um trabalho de transcrição com o particionamento de alto-falantes ativado.

```
{
  "jobName": "job ID",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "Professional answer."
      }
    ],
    "speaker_labels": {
      "speakers": 1,
      "segments": [
        {
          "start_time": "0.000000",
          "speaker_label": "spk_0",
          "end_time": "1.430",
          "items": [
            {
              "start_time": "0.100",
              "speaker_label": "spk_0",
              "end_time": "0.690"
            },
            {
              "start_time": "0.690",
              "speaker_label": "spk_0",
              "end_time": "1.210"
            }
          ]
        }
      ]
    }
  }
}
```

```
    ]
  }
]
},
"items": [
  {
    "start_time": "0.100",
    "end_time": "0.690",
    "alternatives": [
      {
        "confidence": "0.8162",
        "content": "Professional"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "0.690",
    "end_time": "1.210",
    "alternatives": [
      {
        "confidence": "0.9939",
        "content": "answer"
      }
    ],
    "type": "pronunciation"
  },
  {
    "alternatives": [
      {
        "content": "."
      }
    ],
    "type": "punctuation"
  }
]
},
"status": "COMPLETED"
}
```

AWS CLI

Para transcrever um arquivo de áudio de uma conversa entre um médico que pratica cuidados primários e um paciente (AWS CLI)

- Execute o código a seguir.

```
aws transcribe start-transcription-job \  
--region us-west-2 \  
--cli-input-json file://example-start-command.json
```

O código a seguir mostra o conteúdo de `example-start-command.json`.

```
{  
  "MedicalTranscriptionJobName": "my-first-med-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Specialty": "PRIMARYCARE",  
  "Type": "CONVERSATION",  
  "Settings": {  
    "ShowSpeakerLabels": true,  
    "MaxSpeakerLabels": 2  
  }  
}
```

Habilitando o particionamento de alto-falantes em streams em tempo real

Para particionar os alto-falantes e rotular sua fala em um stream em tempo real, use o AWS Management Console ou uma solicitação de streaming. O particionamento de alto-falantes funciona melhor para entre dois e cinco alto-falantes em um stream. Embora o Amazon Transcribe Medical possa particionar mais de cinco alto-falantes em um stream, a precisão das partições diminui se você exceder esse número.

Para iniciar uma solicitação HTTP/2, use a [StartMedicalStreamTranscriptionAPI](#). Para iniciar uma WebSocket solicitação, use um URI pré-assinado. O URI contém as informações necessárias para a configuração da comunicação bidirecional entre o aplicativo e o Amazon Transcribe Médico.

Habilitando o particionamento de alto-falantes em áudio que é falado em seu microfone (AWS Management Console)

Você pode usar o AWS Management Console para iniciar uma transmissão em tempo real de uma conversa médico-paciente ou um ditado que é falado em seu microfone em tempo real.

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, para Amazon Transcribe Medicina, escolha Transcrição em tempo real.
3. Em Tipo de entrada de áudio, escolha o tipo de discurso médico que você deseja transcrever.
4. Para Configurações adicionais, escolha Particionamento de alto-falantes.
5. Escolha Iniciar streaming para começar a transcrever seu áudio em tempo real.
6. Fale no microfone.

Habilitar o particionamento de alto-falantes em um stream HTTP/2

Para ativar o particionamento de alto-falantes em um fluxo HTTP/2 de uma conversa médica, use a [StartMedicalStreamTranscriptionAPI](#) e especifique o seguinte:

- Para `LanguageCode`, especifique o código do idioma que corresponde ao idioma no stream. O valor válido é `en-US`.
- Para `MediaSampleHertz`, especifique a taxa de amostragem do áudio.
- Para `Specialty`, especifique a especialidade médica do provedor.
- `ShowSpeakerLabel – true`

Para obter mais informações sobre como configurar um stream HTTP/2 para transcrever uma conversa médica, consulte [Configurar um fluxo de HTTP/2](#).

Habilitando o particionamento de alto-falantes em uma WebSocket solicitação

Para particionar alto-falantes em WebSocket streams com a API, use o formato a seguir para criar um URI pré-assinado para iniciar uma WebSocket solicitação e `show-speaker-label` definir como `true`.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket
?language-code=languageCode
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&media-encoding=flac
&sample-rate=16000
&session-id=sessionId
&specialty=medicalSpecialty
&type=CONVERSATION
&vocabulary-name=vocabularyName
&show-speaker-label=boolean
```

O código a seguir mostra o exemplo de resposta truncada de uma solicitação de streaming.

```
{
  "Transcript": {
    "Results": [
      {
        "Alternatives": [
          {
            "Items": [
              {
                "Confidence": 0.97,
                "Content": "From",
                "EndTime": 18.98,
                "Speaker": "0",
                "StartTime": 18.74,
                "Type": "pronunciation",
                "VocabularyFilterMatch": false
              },
              {
                "Confidence": 1,
                "Content": "the",
```

```
    "EndTime": 19.31,  
    "Speaker": "0",  
    "StartTime": 19,  
    "Type": "pronunciation",  
    "VocabularyFilterMatch": false  
  },  
  {  
    "Confidence": 1,  
    "Content": "last",  
    "EndTime": 19.86,  
    "Speaker": "0",  
    "StartTime": 19.32,  
    "Type": "pronunciation",  
    "VocabularyFilterMatch": false  
  },  
  ...  
  {  
    "Confidence": 1,  
    "Content": "chronic",  
    "EndTime": 22.55,  
    "Speaker": "0",  
    "StartTime": 21.97,  
    "Type": "pronunciation",  
    "VocabularyFilterMatch": false  
  },  
  ...  
    "Confidence": 1,  
    "Content": "fatigue",  
    "EndTime": 24.42,  
    "Speaker": "0",  
    "StartTime": 23.95,  
    "Type": "pronunciation",  
    "VocabularyFilterMatch": false  
  },  
  {  
    "EndTime": 25.22,  
    "StartTime": 25.22,  
    "Type": "speaker-change",  
    "VocabularyFilterMatch": false  
  },  
  {  
    "Confidence": 0.99,  
    "Content": "True",  
    "EndTime": 25.63,
```

```

        "Speaker": "1",
        "StartTime": 25.22,
        "Type": "pronunciation",
        "VocabularyFilterMatch": false
    },
    {
        "Content": ".",
        "EndTime": 25.63,
        "StartTime": 25.63,
        "Type": "punctuation",
        "VocabularyFilterMatch": false
    }
],
    "Transcript": "From the last note she still has mild sleep deprivation and
chronic fatigue True."
}
],
"EndTime": 25.63,
"IsPartial": false,
"ResultId": "XXXXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
"StartTime": 18.74
}
]
}
}

```

Amazon TranscribeO Medical interrompe o fluxo de áudio de entrada com base em segmentos naturais da fala, como uma mudança no alto-falante ou uma pausa no áudio. A transcrição é retornada progressivamente para seu aplicativo, com cada resposta contendo mais fala transcrita até que todo o segmento seja transcrito. O código anterior é um exemplo truncado de um segmento de fala totalmente transcrito. As etiquetas dos alto-falantes aparecem somente em segmentos totalmente transcritos.

A lista a seguir mostra a organização dos objetos e parâmetros em uma saída de transcrição de streaming.

Transcript

Cada segmento de fala tem seu próprio `Transcript` objeto.

Results

Cada `Transcript` objeto tem seu próprio `Results` objeto. Esse objeto contém o `isPartial` campo. Quando seu valor é `false`, os resultados retornados são para um segmento de fala inteiro.

Alternatives

Cada `Results` objeto tem um `Alternatives` objeto.

Items

Cada `Alternatives` objeto tem seu próprio `Items` objeto que contém informações sobre cada palavra e sinal de pontuação na saída da transcrição. Quando você ativa o particionamento de alto-falantes, cada palavra tem um `Speaker` rótulo para segmentos de fala totalmente transcritos. Amazon Transcribe O Medical usa esse rótulo para atribuir um número inteiro exclusivo a cada alto-falante no stream. O `Type` parâmetro com um valor de `speaker-change` indica que uma pessoa parou de falar e que outra está prestes a começar.

Transcript

Cada objeto `Items` contém um segmento de fala transcrito como valor do `Transcript` campo.

Para obter mais informações sobre WebSocket solicitações, consulte [Configurando um WebSocket stream](#).

Transcrevendo áudio multicanal

Se você tiver um arquivo ou stream de áudio com vários canais, poderá usar a identificação do canal para transcrever a fala de cada um desses canais. Amazon Transcribe O médico transcreve o discurso de cada canal separadamente. Ele combina as transcrições separadas de cada canal em uma única saída de transcrição.

Use a identificação de canais para identificar os canais separados em seu áudio e transcrever a fala de cada um desses canais. Habilite isso em situações como um cenário de chamador e agente. Use isso para distinguir um chamador de um agente em gravações ou transmissões de centros de contato que realizam o monitoramento da segurança de medicamentos.

Você pode ativar a identificação de canais para processamento em lote e streaming em tempo real. A lista a seguir descreve como habilitá-la para cada método.

- Transcrição Batch —AWS Management Console e [StartMedicalTranscriptionJobAPI](#)
- Transcrição de WebSocket streaming — streaming e [StartMedicalStreamTranscriptionAPI](#)

Transcrevendo arquivos de áudio multicanal

Quando você transcreve um arquivo de áudio, oAmazon Transcribe Medical retorna uma lista de itens para cada canal. Um item é uma palavra transcrita ou um sinal de pontuação. Cada palavra tem uma hora de início e uma hora de término. Se uma pessoa em um canal falar sobre outra pessoa em um canal separado, os horários de início e término dos itens de cada canal se sobrepõem enquanto os indivíduos falam uns sobre os outros.

Por padrão, você pode transcrever arquivos de áudio com dois canais. Você pode solicitar um aumento de cota se precisar transcrever arquivos que tenham mais de dois canais. Para obter informações sobre como solicitar um aumento de cota, consulte [AWS service \(Serviço da AWS\)cotas](#).

Para transcrever áudio multicanal em um trabalho de transcrição em lote, use oAWS Management Console ou a [StartMedicalTranscriptionJobAPI](#).

AWS Management Console

Para usar oAWS Management Console para habilitar a identificação do canal em seu trabalho de transcrição em lote, você ativa a identificação de áudio e, em seguida, a identificação do canal. A identificação do canal é um subconjunto da identificação de áudio noAWS Management Console.

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, emAmazon Transcribe Principal, escolha Trabalhos de transcrição.
3. Escolha Create job (Criar trabalho).
4. Na página Especificar detalhes do trabalho, forneça informações sobre seu trabalho de transcrição.
5. Escolha Próximo.
6. Ative a identificação de áudio.
7. Para Tipo de identificação de áudio, escolha Identificação de canal.
8. Escolha Create (Criar).

API

Para transcrever um arquivo de áudio multicanal (API)

- Para a [StartMedicalTranscriptionJob](#) API, especifique o seguinte.
 - a. Para `TranscriptionJobName`, especifique um nome exclusivo para sua Conta da AWS.
 - b. Para `LanguageCode`, especifique o código do idioma que corresponde ao idioma falado no arquivo de áudio. O valor válido é `en-US`.
 - c. Para o `MediaFileUri` parâmetro do `Media` objeto, especifique o nome do arquivo de mídia que você deseja transcrever.
 - d. Para o `Settings` objeto, `ChannelIdentification` defina como `true`.

A seguir está um exemplo de solicitação usando AWS SDK for Python (Boto3) o.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_name = "my-first-med-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION',
    Settings = {
        'ChannelIdentification': True
    }
)
while True:
    status = transcribe.get_transcription_job(MedicalTranscriptionJobName = job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
        'FAILED']:
```

```
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

AWS CLI

Para transcrever um arquivo de áudio multicanal usando uma tarefa de transcrição em lote (AWS CLI)

- Execute o código a seguir.

```
aws transcribe start-medical-transcription-job \  
--region us-west-2 \  
--cli-input-json file://example-start-command.json
```

A seguir está o código de `example-start-command.json`.

```
{
  "MedicalTranscriptionJobName": "my-first-med-transcription-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "OutputKey": "my-output-files/",
  "LanguageCode": "en-US",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",

  "Settings":{
    "ChannelIdentification": true
  }
}
```

O código a seguir mostra a saída da transcrição de um arquivo de áudio que tem uma conversa em dois canais.

```
{
  "jobName": "job id",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "When you try ... It seems to ..."
      }
    ],
    "channel_labels": {
      "channels": [
        {
          "channel_label": "ch_0",
          "items": [
            {
              "start_time": "12.282",
              "end_time": "12.592",
              "alternatives": [
                {
                  "confidence": "1.0000",
                  "content": "When"
                }
              ],
              "type": "pronunciation"
            },
            {
              "start_time": "12.592",
              "end_time": "12.692",
              "alternatives": [
                {
                  "confidence": "0.8787",
                  "content": "you"
                }
              ],
              "type": "pronunciation"
            },
            {
              "start_time": "12.702",
              "end_time": "13.252",
              "alternatives": [
                {
                  "confidence": "0.8318",
```

```
        "content": "try"
      }
    ],
    "type": "pronunciation"
  },
  ...
]
},
{
  "channel_label": "ch_1",
  "items": [
    {
      "start_time": "12.379",
      "end_time": "12.589",
      "alternatives": [
        {
          "confidence": "0.5645",
          "content": "It"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "12.599",
      "end_time": "12.659",
      "alternatives": [
        {
          "confidence": "0.2907",
          "content": "seems"
        }
      ],
      "type": "pronunciation"
    },
    {
      "start_time": "12.669",
      "end_time": "13.029",
      "alternatives": [
        {
          "confidence": "0.2497",
          "content": "to"
        }
      ],
      "type": "pronunciation"
    }
  ],
}
```

```
    ]  
  }  
}
```

Transcrevendo fluxos de áudio multicanal

Você pode transcrever áudio de canais separados em HTTP/2 ou WebSocket streams usando a [StartMedicalStreamTranscriptionAPI](#).

Por padrão, você pode transcrever streams com dois canais. Você pode solicitar um aumento de cota se precisar transcrever streams que tenham mais de dois canais. Para obter informações sobre como solicitar um aumento de cota, consulte [cotasAWS de serviço](#).

Transcrevendo áudio multicanal em um stream HTTP/2

Para transcrever áudio multicanal em um stream HTTP/2, use a [StartMedicalStreamTranscriptionAPI](#) e especifique o seguinte:

- `LanguageCode`: O código de idioma do áudio. O valor válido é `en-US`.
- `MediaEncoding`— A codificação do áudio. Os valores válidos são `ogg-opus`, `flac` e `pcm`.
- `EnableChannelIdentification` – `true`
- `NumberOfChannels`: o número de canais em seu streaming de áudio.

Para obter mais informações sobre como configurar um stream HTTP/2 para transcrever uma conversa médica, consulte [Configurar um fluxo de HTTP/2](#).

Transcrevendo áudio multicanal em um WebSocket stream

Para particionar alto-falantes em WebSocket streams, use o formato a seguir para criar um URI pré-assinado e iniciar uma WebSocket solicitação. Especifique `enable-channel-identification` como `true` e o número de canais em sua transmissão em `number-of-channels`. Um URI pré-assinado contém as informações necessárias para configurar a comunicação bidirecional entre seu aplicativo e o Amazon Transcribe Medical.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-  
transcription-websocket  
?language-code=LanguageCode  
&X-Amz-Algorithm=AWS4-HMAC-SHA256
```

```
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-  
west-2%2Ftranscribe%2Faws4_request  
&X-Amz-Date=20220208T235959Z  
&X-Amz-Expires=300  
&X-Amz-Security-Token=security-token  
&X-Amz-Signature=Signature Version 4 signature  
&X-Amz-SignedHeaders=host  
&media-encoding=flac  
&sample-rate=16000  
&session-id=sessionId  
&enable-channel-identification=true  
&number-of-channels=2
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Para obter mais informações sobre WebSocket solicitações, consulte [Configurando um WebSocket stream](#).

Saída de streaming multicanal

A saída de uma transcrição de streaming é a mesma para HTTP/2 e WebSocket solicitações. Veja a seguir um exemplo de saída.

```
{  
  "resultId": "XXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",  
  "startTime": 0.11,  
  "endTime": 0.66,  
  "isPartial": false,  
  "alternatives": [  
    {  
      "transcript": "Left.",  
      "items": [  
        {  
          "startTime": 0.11,  
          "endTime": 0.45,  
          "type": "pronunciation",  
          "content": "Left",  
          "vocabularyFilterMatch": false  
        },  
        {  
          "startTime": 0.45,
```



```
        "endTime": 0.45,  
        "type": "punctuation",  
        "content": ".",  
        "vocabularyFilterMatch": false  
      }  
    ]  
  },  
  ],  
  "channelId": "ch_0"  
}
```

Para cada segmento de fala, há um `channelId` bandeira que indica a qual canal a fala pertence.

Transcrevendo um ditado médico

Você pode usar o Amazon Transcribe Medical para transcrever anotações médicas ditadas pelo médico usando um trabalho de transcrição em lote ou um stream em tempo real. Os trabalhos de transcrição em Batch permitem que você transcreva arquivos de áudio. Você especifica a especialidade médica do médico em seu trabalho ou fluxo de transcrição para garantir que o Amazon Transcribe Medical produza resultados de transcrição com a maior precisão possível.

Você pode transcrever um ditado médico nas seguintes especialidades:

- Cardiologia — disponível somente na transcrição em streaming
- Neurologia — disponível somente na transcrição de streaming
- Oncologia — disponível somente na transcrição em streaming
- Atenção primária — inclui os seguintes tipos de prática médica:
 - Medicina da família
 - Medicina interna
 - Obstetrícia e ginecologia (OB-GYN)
 - Pediatria
- Radiologia — disponível somente na transcrição por streaming
- Urologia — disponível somente na transcrição em streaming

Você pode melhorar a precisão da transcrição usando vocabulários personalizados. Para obter informações sobre como funcionam os vocabulários médicos personalizados, consulte [Melhorando a precisão da transcrição com vocabulários médicos personalizados](#).

Por padrão, o Amazon Transcribe Medical retorna a transcrição com o nível de confiança mais alto. Se você quiser configurá-lo para retornar transcrições alternativas, consulte [Gerando transcrições alternativas](#).

Para obter informações sobre como os números e as medidas médicas aparecem na saída da transcrição, consulte [Transcrevendo números](#) [Transcrevendo termos e medidas médicas](#) e.

Tópicos

- [Transcrevendo um arquivo de áudio de um ditado médico](#)
- [Transcrevendo um ditado médico em um stream em tempo real](#)

Transcrevendo um arquivo de áudio de um ditado médico

Use um trabalho de transcrição em lote para transcrever arquivos de áudio de conversas médicas. Você pode usar isso para transcrever um diálogo médico-paciente. Você pode iniciar um trabalho de transcrição em lote na [StartMedicalTranscriptionJob](#) API ou no AWS Management Console.

Ao iniciar um trabalho de transcrição médica com a [StartMedicalTranscriptionJob](#) API, você especifica PRIMARYCARE como o valor do Specialty parâmetro.

AWS Management Console

Para transcrever um diálogo médico-paciente (AWS Management Console)

Para usar o diálogo AWS Management Console para transcrever um diálogo médico-paciente, crie um trabalho de transcrição e escolha Conversa para o tipo de entrada de áudio.

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, em Amazon Transcribe Médico, escolha Tâncias programadas.
3. Escolha Create job (Criar trabalho).
4. Na página Especificar detalhes do Job, em Configurações do trabalho, especifique o seguinte.
 - a. Nome — o nome do trabalho de transcrição.
 - b. Tipo de entrada de áudio — Ditado
5. Para os campos restantes, especifique a Amazon S3 localização do seu arquivo de áudio e onde você deseja armazenar a saída do trabalho de transcrição.
6. Escolha Next (Próximo).
7. Escolha Create (Criar).

API

Para transcrever uma conversa médica usando um trabalho de transcrição em lote (API)

- Para a [StartMedicalTranscriptionJob](#) API, especifique o seguinte.
 - a. Para `MedicalTranscriptionJobName`, especifique um nome exclusivo em sua conta da AWS.
 - b. Para `LanguageCode`, especifique o código do idioma que corresponde ao idioma falado no seu arquivo de áudio e ao idioma do seu filtro de vocabulário.
 - c. No `MediaFileUri` parâmetro do `Media` objeto, especifique o nome do arquivo de áudio que você deseja transcrever.
 - d. Para `Specialty`, especifique a especialidade médica do médico que fala no arquivo de áudio.
 - e. Para `Type`, especifique `DICTATION`.
 - f. Para `OutputBucketName`, especifique o Amazon S3 bucket para armazenar os resultados da transcrição.

A seguir está um exemplo de solicitação que usa o AWS SDK for Python (Boto3) para transcrever um ditado médico de um médico da `PRIMARYCARE` especialidade.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe')
job_name = "my-first-med-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'DICTATION'
)
```

```
while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

O código de exemplo a seguir mostra os resultados da transcrição de um ditado médico.

```
{
  "jobName": "dictation-medical-transcription-job",
  "accountId": "111122223333",
  "results": {
    "transcripts": [
      {
        "transcript": "... came for a follow up visit today..."
      }
    ],
    "items": [
      {
        ...
        "start_time": "4.85",
        "end_time": "5.12",
        "alternatives": [
          {
            "confidence": "1.0",
            "content": "came"
          }
        ],
        "type": "pronunciation"
      },
      {
        "start_time": "5.12",
        "end_time": "5.29",
        "alternatives": [
          {
            "confidence": "1.0",
```

```
        "content": "for"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.29",
    "end_time": "5.33",
    "alternatives": [
      {
        "confidence": "0.9955",
        "content": "a"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.33",
    "end_time": "5.66",
    "alternatives": [
      {
        "confidence": "0.9754",
        "content": "follow"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.66",
    "end_time": "5.75",
    "alternatives": [
      {
        "confidence": "0.9754",
        "content": "up"
      }
    ],
    "type": "pronunciation"
  },
  {
    "start_time": "5.75",
    "end_time": "6.02",
    "alternatives": [
      {
        "confidence": "1.0",
```

```
        "content": "visit"
      }
    ]
    ...
  },
  "status": "COMPLETED"
}
```

AWS CLI

Para habilitar o particionamento de alto-falantes em um trabalho de transcrição em lote (AWS CLI)

- Execute o código a seguir.

```
aws transcribe start-medical-transcription-job \  
--region us-west-2 \  
--cli-input-json file://example-start-command.json
```

O código a seguir mostra o conteúdo de `example-start-command.json`.

```
{  
  "MedicalTranscriptionJobName": "my-first-med-transcription-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "OutputKey": "my-output-files/",  
  "LanguageCode": "en-US",  
  "Specialty": "PRIMARYCARE",  
  "Type": "DICTATION"  
}
```

Transcrevendo um ditado médico em um stream em tempo real

Use um WebSocket stream para transcrever um ditado médico como um fluxo de áudio. Você também pode usar o AWS Management Console para transcrever a fala que você ou outras pessoas falam diretamente em um microfone.

Para um HTTP/2 ou um WebSocket stream, você pode transcrever áudio nas seguintes especialidades médicas:

- Cardiologia
- Oncologia
- Neurologia
- Primary Care
- Radiologia
- Urologia

Cada especialidade médica inclui muitos tipos de procedimentos e consultas. Os médicos, portanto, escolha muitos tipos diferentes de notas. Use os exemplos a seguir como orientação para ajudá-lo a especificar o valor do parâmetro `specialty` URI da WebSocket solicitação ou o `specialty` parâmetro da [StartMedicalStreamTranscriptionAPI](#):

- Para um ditado após o procedimento de eletrofisiologia ou ecocardiograma, escolha `CARDIOLOGY`.
- Para um ditado após um procedimento cirúrgico de oncologia ou oncologia por radiação, escolha `ONCOLOGY`.
- Para um médico ditando notas indicando um diagnóstico de encefalite, escolha `NEUROLOGY`.
- Para um ditado de notas de procedimento para quebrar uma pedra na bexiga, escolha `UROLOGY`.
- Para um ditado das anotações do médico após uma consulta de medicina interna, escolha `PRIMARYCARE`.
- Para um ditado de um médico comunicando os resultados de uma tomografia computadorizada, PET, ressonância magnética ou radiografia, escolha `RADIOLOGY`.
- Para um ditado de anotações médicas após uma consulta de ginecologia, escolha `PRIMARYCARE`.

Para melhorar a precisão da transcrição de termos específicos em um stream em tempo real, use um vocabulário personalizado. Para ativar um vocabulário personalizado, defina o valor `devocabulary-name` como o nome do vocabulário personalizado que você deseja usar.

Transcrevendo um ditado falado em seu microfone com oAWS Management Console

Para usar oAWS Management Console para transcrever o streaming de áudio de um ditado médico, escolha a opção de transcrever um ditado médico, iniciar a transmissão e começar a falar no microfone.

Para transcrever o streaming de áudio de um ditado médico (AWS Management Console)

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, emAmazon Transcribe Médico, escolha Transcrições programadas.
3. Escolha Ditado.
4. Para especialidade médica, escolha a especialidade médica do médico que fala no stream.
5. Selecione Start streaming (Iniciar transmissão).
6. Fale no microfone.

Transcrevendo um ditado em um stream do HTTP/2 do

Para transcrever um fluxo HTTP/2 de um ditado médico, use a [StartMedicalStreamTranscription](#)API e especifique o seguinte:

- `LanguageCode`— O código do idioma. O valor válido éen-US
- `MediaEncoding`— A codificação usada para o áudio de entrada. Os valores válidos são pcm, ogg-opus e flac.
- `Specialty`— A especialidade do profissional médico.
- `Type` – DICTATION

Para obter mais informações sobre como configurar um fluxo HTTP/2 para transcrever um ditado médico, consulte[Configurar um fluxo de HTTP/2](#).

Usando uma solicitação WebSocket de streaming para transcrever um ditado médico

Para transcrever um ditado médico em um stream em tempo real usando uma WebSocket solicitação, você cria um URI pré-assinado. Esse URI contém as informações necessárias para configurar o fluxo de áudio entre seu aplicativo e oAmazon Transcribe Medical. Para obter mais informações sobre a criação WebSocket de solicitações, consulte[Configurando um WebSocket stream](#).

Use o modelo a seguir para criar seu URI para criar seu URI.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket
?language-code=LanguageCode
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&media-encoding=flac
&sample-rate=16000
&session-id=sessionId
&specialty=medicalSpecialty
&type=DICTATION
&vocabulary-name=vocabularyName
&show-speaker-label=boolean
```

Para obter mais informações sobre a criação de URIs pré-assinados, consulte [Configurando um WebSocket stream](#).

Melhorando a precisão da transcrição com vocabulários médicos personalizados

Para melhorar a precisão da transcrição em Amazon Transcribe Medicina, crie e use um ou mais vocabulários médicos personalizados. Um vocabulário personalizado é uma coleção de palavras ou frases específicas de um domínio. Esta coleção ajuda a melhorar o desempenho da Amazon Transcribe Medicina na transcrição dessas palavras ou frases.

Você é responsável pela integridade de seus próprios dados ao usar o Amazon Transcribe Medical. Não insira informações confidenciais, informações pessoais (PII) ou informações de saúde protegidas (PHI) em um vocabulário personalizado.

Para obter melhores resultados, crie pequenos vocabulários personalizados separados, cada um ajudando a transcrever uma gravação de áudio específica. Você obtém maiores melhorias na precisão da transcrição do que se tivesse criado um grande vocabulário personalizado para usar com todas as suas gravações.

Por padrão, você pode ter até 100 vocabulários personalizados em sua Conta da AWS. Um vocabulário personalizado não pode exceder 50 KB de tamanho. Para obter informações sobre como solicitar um aumento no número de vocabulários personalizados que você pode ter em sua Conta da AWS, consulte [Cotas AWS de serviço](#).

Vocabulários personalizados estão disponíveis em inglês americano (en-US).

Tópicos

- [Criando um arquivo de texto para seu vocabulário médico personalizado](#)
- [Usando um arquivo de texto para criar um vocabulário médico personalizado](#)
- [Transcrevendo um arquivo de áudio usando um vocabulário médico personalizado](#)
- [Transcrevendo um stream em tempo real usando um vocabulário médico personalizado](#)
- [Conjunto de caracteres para Amazon Transcribe Medicina](#)

Criando um arquivo de texto para seu vocabulário médico personalizado

Para criar um vocabulário personalizado, você cria um arquivo de texto no formato UTF-8. Nesse arquivo, você cria uma tabela de quatro colunas, com cada coluna especificando um campo. Cada campo informa à Amazon Transcribe Medicina como os termos específicos do domínio são pronunciados ou como exibi-los em suas transcrições. Você armazena o arquivo de texto contendo esses campos em um Amazon S3 bucket.

Entendendo como formatar seu arquivo de texto

Para criar um vocabulário médico personalizado, você insere os nomes das colunas como uma linha de cabeçalho. Você insere os valores de cada coluna abaixo da linha do cabeçalho.

A seguir estão os nomes das quatro colunas da tabela:

- `Phrase`— coluna obrigatória, valores obrigatórios
- `IPA`— coluna obrigatória, os valores podem ser opcionais
- `SoundsLike`— coluna obrigatória, os valores podem ser opcionais
- `DisplayAs`— coluna obrigatória, os valores podem ser opcionais

Ao criar um vocabulário personalizado, certifique-se de:

- Separe cada coluna com um único caractere Tab. Amazon Transcribe emite uma mensagem de erro se você tentar separar as colunas com espaços ou caracteres de tabulação.
- Certifique-se de que não haja espaços à direita ou espaços em branco após cada valor em uma coluna.

Certifique-se de que os valores inseridos para cada coluna:

- Tenha menos de 256 caracteres, incluindo hífen
- Use somente caracteres do conjunto de caracteres permitido, consulte [Conjunto de caracteres para Amazon Transcribe Medicina](#).

Inserindo valores para as colunas da tabela

As informações a seguir mostram como especificar valores para as quatro colunas da tabela:

- **Phrase**— A palavra ou frase que deve ser reconhecida. Você deve inserir valores nesta coluna.

Se a entrada for uma frase, separe as palavras com um hífen (-). Por exemplo, insira **cerebral autosomal dominant arteriopathy with subcortical infarcts and leukoencephalopathy** como **cerebral-autosomal-dominant-arteriopathy-with-subcortical-infarcts-and-leukoencephalopathy**.

Insira os acrônimos ou outras palavras cujas letras devem ser pronunciadas individualmente como letras únicas seguidas por pontos, como **D.N.A.** ou **S.T.E.M.I.**. Para inserir a forma plural de um acrônimo, como "STEMIs", separe o "s" da sigla com um hífen: "**S.T.E.M.I-s**" Você pode usar letras maiúsculas ou minúsculas para acrônimos.

A coluna **Phrase** é obrigatória. Você pode usar qualquer um dos caracteres permitidos para o idioma de entrada. Para caracteres permitidos, consulte [Conjunto de caracteres para Amazon Transcribe Medicina](#). Se você não especificar a **DisplayAs** coluna, o Amazon Transcribe Medical usará o conteúdo da **Phrase** coluna no arquivo de saída.

- **IPA**(coluna obrigatória, os valores podem ser opcionais) — Para especificar a pronúncia de uma palavra ou frase, você pode incluir caracteres no [Alfabeto Fonético Internacional \(IPA\)](#) nessa coluna. A coluna IPA não pode conter espaços no início nem no final, e você deve usar um único espaço para separar cada fonema na entrada. Por exemplo, em inglês, a expressão **acute-respiratory-distress-syndrome** seria inserida como

`ə k j u t # # s p # # ə t # # i d # s t # # s s # n d # o # m`. A expressão **A.L.L.** seria inserida como `e # # l # l`.

Mesmo que não especifique o conteúdo da coluna IPA, você deve incluir uma coluna IPA em branco. Se você incluir valores na coluna IPA, não poderá fornecer valores para a coluna SoundsLike.

Para obter uma lista de caracteres IPA permitidos para um idioma específico, consulte [Conjunto de caracteres para Amazon Transcribe Medicina](#). O inglês americano é o único idioma disponível em Amazon Transcribe Medicina.

- **SoundsLike**(coluna obrigatória, os valores podem ser opcionais) — Você pode dividir uma palavra ou frase em segmentos menores e fornecer uma pronúncia para cada segmento usando a ortografia padrão do idioma para imitar a forma como a palavra soa. Por exemplo, em inglês, você pode fornecer dicas de pronúncia para a expressão **cerebral-autosomal-dominant-arteriopathy-with-subcortical-infarcts-and-leukoencephalopathy** desta forma: **sir-e-brul-aut-o-som-ul-dah-mi-nant-ar-ter-ri-o-pa-ty-with-sub-cor-ti-cul-in-farcts-and-lewk-o-en-ce-phul-ah-pu-ty**. A dica para a expressão **atrioventricular-nodal-reentrant-tachycardia** seria assim: **ay-tree-o-ven-trick-u-lar-node-al-re-entr-ant-tack-ih-card-ia**. Você separa cada parte da dica com um hífen (-).

Mesmo que não forneça valores para a coluna SoundsLike, você deve incluir uma coluna SoundsLike em branco. Se você incluir valores na coluna SoundsLike, não poderá fornecer valores para a coluna IPA.

Você pode usar qualquer um dos caracteres permitidos para o idioma de entrada. Para obter uma lista de caracteres permitidos, consulte [Conjunto de caracteres para Amazon Transcribe Medicina](#).

- **DisplayAs**(coluna obrigatória, os valores podem ser opcionais) — Define a aparência da palavra ou frase quando é emitida. Por exemplo, se a palavra ou expressão for **cerebral-autosomal-dominant-arteriopathy-with-subcortical-infarcts-and-leukoencephalopathy**, você poderá especificar a forma de exibição como `cerebral autosomal dominant arteriopathy with subcortical infarcts and leukoencephalopathy`, para que o hífen não esteja presente. Você também poderá especificar `DisplayAs` como `CADASIL` se desejar mostrar o acrônimo em vez do termo completo na saída.

Se você não especificar a `DisplayAs` coluna, o Amazon Transcribe Medical usará a `Phrase` coluna do arquivo de entrada na saída.

Você pode usar qualquer caractere UTF-8 na coluna `DisplayAs`.

Você pode incluir espaços somente para os valores nas `DisplayAs` colunas IPA e `IPA`.

Para criar o arquivo de texto do seu vocabulário personalizado, coloque cada palavra ou frase em seu arquivo de texto em uma linha separada. Separe as colunas com caracteres de tabulação. Inclua espaços somente para valores nas colunas IPA e `DisplayAs`. Salve o arquivo com a extensão `.txt` em um Amazon S3 bucket no mesmo Região da AWS local em que você usa o Amazon Transcribe Medical para criar seu vocabulário personalizado.

Se você editar o arquivo de texto no Windows, certifique-se de que seu arquivo esteja no formato LF e não no formato CRLF. Caso contrário, você não poderá criar seu vocabulário personalizado. Alguns editores de texto permitem alterar a formatação com os comandos Find (Localizar) e Replace (Substituir).

Os exemplos a seguir mostram o texto que você pode usar para criar vocabulários personalizados. Para criar um vocabulário personalizado a partir destes exemplos, copie um exemplo em um editor de texto, substitua `[TAB]` por um caractere de tabulação e carregue o arquivo de texto salvo no Amazon S3.

```
Phrase[TAB]IPA[TAB]SoundsLike[TAB]DisplayAs
acute-respiratory-distress-syndrome[TAB][TAB][TAB]acute respiratory distress syndrome
A.L.L.[TAB]e# # 1 # 1[TAB][TAB]ALL
atrioventricular-nodal-reentrant-tachycardia[TAB][TAB]ay-tree-o-ven-trick-u-lar-node-
al-re-entr-ant-tack-ih-card-ia[TAB]
```

Você pode inserir colunas em qualquer ordem. Os exemplos a seguir mostram outras estruturas válidas para o arquivo de entrada de vocabulário personalizado.

```
Phrase[TAB]SoundsLike[TAB]IPA[TAB]DisplayAs
acute-respiratory-distress-syndrome[TAB][TAB][TAB]acute respiratory distress syndrome
A.L.L.[TAB][TAB]e# # 1 # 1[TAB]ALL
atrioventricular-nodal-reentrant-tachycardia[TAB]ay-tree-o-ven-trick-u-lar-node-al-re-
entr-ant-tack-ih-card-ia[TAB][TAB]
```

```
DisplayAs[TAB]SoundsLike[TAB]IPA[TAB]Phrase
acute respiratory distress syndrome[TAB][TAB][TAB]acute-respiratory-distress-syndrome
```

```
ALL[TAB][TAB]e# # 1 # 1[TAB]A.L.L.
[TAB]ay-tree-o-ven-trick-u-lar-node-al-re-entr-ant-tack-ih-card-ia[TAB]
[TAB]atrioventricular-nodal-reentrant-tachycardia
```

Para facilitar a leitura, as tabelas a seguir mostram os exemplos anteriores mais claramente no formato html. Eles servem apenas para ilustrar os exemplos.

Phrase	AFI	SoundsLike	DisplayAs
acute-respiratory-distress-syndrome			acute respiratory distress syndrome
A.L.L.	eɪ ɛ ɛ		ALL
atrioventricular-nodal-reentrant-tachycardia		ay-tree-o-ven-trick-ular-node-al-re-entr-ant-tack-ih-card-ia	

Phrase	SoundsLike	AFI	DisplayAs
acute-respiratory-distress-syndrome			acute respiratory distress syndrome
atrioventricular-nodal-reentrant-tachycardia	ay-tree-o-ven-trick-ular-node-al-re-entr-ant-tack-ih-card-ia		
A.L.L.		eɪ ɛ ɛ	ALL

DisplayAs	SoundsLike	AFI	Phrase
acute respiratory distress syndrome			acute-respiratory-distress-syndrome
ALL		eɪ ɛ ɛ	A.L.L.

DisplayAs	SoundsLike	AFI	Phrase
	ay-tree-o-ven-trick-u-lar-node-al-re-entr-ant-tack-ih-card-ia		atrioventricular-nodal-reentrant-tachycardia

Usando um arquivo de texto para criar um vocabulário médico personalizado

Para criar um vocabulário personalizado, você deve ter preparado um arquivo de texto que contenha uma coleção de palavras ou frases. Amazon Transcribe O Medical usa esse arquivo de texto para criar um vocabulário personalizado que pode ser usado para melhorar a precisão da transcrição dessas palavras ou frases. Você pode criar um vocabulário personalizado usando a [CreateMedicalVocabulary](#) API ou o console Amazon Transcribe médico.

AWS Management Console

Para usar o AWS Management Console para criar um vocabulário personalizado, você fornece o Amazon S3 URI do arquivo de texto contendo suas palavras ou frases.

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, em Amazon Transcribe Médico, escolha Vocabulário personalizado.
3. Em Nome, em Configurações de vocabulário, escolha um nome para seu vocabulário personalizado.
4. Especifique a localização do arquivo de áudio ou arquivo de vídeo no Amazon S3:
 - Para a localização do arquivo de entrada de vocabulário no S3, em Configurações de vocabulário, especifique o Amazon S3 URI que identifica o arquivo de texto que você usará para criar seu vocabulário personalizado.
 - Para Localização do arquivo de entrada de vocabulário no S3, escolha Procurar S3 para procurar o arquivo de texto e selecioná-lo.
5. Escolha Criar vocabulário.

Você pode ver o status de processamento do seu vocabulário personalizado no AWS Management Console.

API

Para criar um vocabulário médico personalizado (API)

- Para a [StartTranscriptionJob](#)API, especifique o seguinte.
 - a. Para `LanguageCode`, especifique `en-US`.
 - b. Para `VocabularyFileUri`, especifique a Amazon S3 localização do arquivo de texto que você usa para definir seu vocabulário personalizado.
 - c. Para `VocabularyName`, especifique um nome para seu vocabulário personalizado. O nome que pode ser especificado deve ser exclusivo em sua Conta da AWS.

Para ver o status de processamento do seu vocabulário personalizado, use a [GetMedicalVocabulary](#)API.

Este é um exemplo de solicitação que usa o AWS SDK for Python (Boto3) para criar um vocabulário personalizado.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
vocab_name = "my-first-vocabulary"
response = transcribe.create_medical_vocabulary(
    VocabularyName = job_name,
    VocabularyFileUri = 's3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-
table.txt'
    LanguageCode = 'en-US',
)

while True:
    status = transcribe.get_medical_vocabulary(VocabularyName = vocab_name)
    if status['VocabularyState'] in ['READY', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```


AWS CLI

Para habilitar o particionamento de alto-falantes em um trabalho de transcrição em lote (AWS CLI)

- Execute o código a seguir.

```
aws transcribe create-medical-vocabulary \  
--vocabulary-name my-first-vocabulary \  
--vocabulary-file-uri s3://DOC-EXAMPLE-BUCKET/my-vocabularies/my-vocabulary-  
file.txt \  
--language-code en-US
```

Transcrevendo um arquivo de áudio usando um vocabulário médico personalizado

Use o [StartMedicalTranscriptionJob](#) ou o AWS Management Console para iniciar um trabalho de transcrição que usa um vocabulário personalizado para melhorar a precisão da transcrição.

AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, em Amazon Transcribe Médico, escolha Tarefas de transcrição.
3. Escolha Create job (Criar trabalho).
4. Na página Especificar detalhes do trabalho, forneça informações sobre seu trabalho de transcrição.
5. Escolha Próximo.
6. Em Personalização, ative Vocabulário personalizado.
7. Em Seleção de vocabulário, escolha um vocabulário personalizado.
8. Escolha Create (Criar).

API

Para habilitar o particionamento de alto-falantes em um arquivo de áudio usando um trabalho de transcrição em lote (API)

- Para a [StartMedicalTranscriptionJob](#) API, especifique o seguinte.

- a. Para `MedicalTranscriptionJobName`, especifique um nome exclusivo em sua Conta da AWS.
- b. Para `LanguageCode`, especifique o código do idioma que corresponde ao idioma falado no seu arquivo de áudio e ao idioma do seu filtro de vocabulário.
- c. Para o `MediaFileUri` parâmetro do `Media` objeto, especifique o nome do arquivo de áudio que você deseja transcrever.
- d. Para `Specialty`, especifique a especialidade médica do médico que fala no arquivo de áudio.
- e. Para `Type`, especifique se o arquivo de áudio é uma conversa ou um ditado.
- f. Para `OutputBucketName`, especifique o Amazon S3 bucket para armazenar os resultados da transcrição.
- g. Para o `Settings` objeto, especifique o seguinte.
 - `VocabularyName`— o nome do seu vocabulário personalizado.

A solicitação a seguir usa o AWS SDK for Python (Boto3) para iniciar um trabalho de transcrição em lote com um vocabulário personalizado.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-med-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION',
    Settings = {
        'VocabularyName': 'example-med-custom-vocab'
    }
)
```

```
)

while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Transcrevendo um stream em tempo real usando um vocabulário médico personalizado

Para melhorar a precisão da transcrição em um stream em tempo real, você pode usar um vocabulário personalizado usando HTTP/2 ou WebSocket streams. Para iniciar uma solicitação HTTP/2, use a [StartMedicalStreamTranscription](#) API. Você pode usar um vocabulário personalizado em tempo real usando a [StartMedicalStreamTranscription](#) API ou usando o WebSocket protocolo. AWS Management Console

Transcrevendo um ditado falado em seu microfone (AWS Management Console)

Para usar o AWS Management Console para transcrever o streaming de áudio de um ditado médico, escolha a opção de transcrever um ditado médico, iniciar a transmissão e começar a falar no microfone.

Para transcrever o streaming de áudio de um ditado médico (AWS Management Console)

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, em Amazon Transcribe Médico, escolha Transcrição em tempo real.
3. Para especialidade médica, escolha a especialidade médica do médico que fala no stream.
4. Para Tipo de entrada de áudio, escolha Conversa ou Ditado.
5. Para Configurações adicionais, escolha Vocabulário personalizado.
 - Para Seleção de vocabulário, escolha o vocabulário personalizado.
6. Selecione Start streaming (Iniciar transmissão).

7. Fale no microfone.

Habilitando o particionamento de alto-falantes em um stream HTTP/2

Veja a seguir a sintaxe dos parâmetros de uma solicitação HTTP/2.

```
POST /medical-stream-transcription HTTP/2
host: transcribestreaming.us-west-2.amazonaws.com
authorization: Generated value
x-amz-target: com.amazonaws.transcribe.Transcribe.StartMedicalStreamTranscription
x-amz-content-sha256: STREAMING-MED-AWS4-HMAC-SHA256-EVENTS
x-amz-date: 20220208T235959Z
x-amzn-transcribe-session-id: my-first-http2-med-stream
x-amzn-transcribe-language-code: en-US
x-amzn-transcribe-media-encoding: flac
x-amzn-transcribe-sample-rate: 16000
x-amzn-transcribe-vocabulary-name: my-first-med-vocab
x-amzn-transcribe-specialty: PRIMARYCARE
x-amzn-transcribe-type: CONVERSATION
x-amzn-transcribe-show-speaker-label: true
Content-type: application/vnd.amazon.eventstream
transfer-encoding: chunked
```

Descrições dos parâmetros:

- host: atualize o Região da AWS ('us-west-2' no exemplo anterior) com o Região da AWS que você está chamando. Para obter uma lista de pontos válidos Regiões da AWS, consulte [Regiões da AWS e Endpoints](#).
- autorização: Esse é um campo gerado. Para saber mais sobre como criar uma assinatura, consulte Assinatura de [AWS solicitações com a versão 4 do Signature](#).
- x-amz-target: Não altere esse campo; use o conteúdo mostrado no exemplo anterior.
- x-amz-content-sha256: Esse é um campo gerado. Para saber mais sobre como calcular uma assinatura, consulte Assinatura de [AWS solicitações com a versão 4 do Signature](#).
- x-amz-date: a data e a hora em que a assinatura é criada. O formato é YYYYMMDDTHHMMSSZ, onde YYYY=ano, MM=mês, DD=dia, HH=hora, MM=minuto, SS=segundos e 'T' e 'Z' são caracteres fixos. Para obter mais informações, consulte [Processamento de datas na versão 4 do Signature](#).
- x-amzn-transcribe-session-id: o nome da sua sessão de streaming.

- `x-amzn-transcribe-language-code`: a codificação usada para seu áudio de entrada. Consulte [StartMedicalStreamTranscription](#) ou [idiomas oferecidos e recursos específicos do idioma](#) para obter uma lista de valores válidos.
- `x-amzn-transcribe-media-encoding`: A codificação usada para o áudio de entrada. Os valores válidos são `pcm`, `ogg-opus` e `flac`.
- `x-amzn-transcribe-sample-rate`: A taxa de amostragem do áudio de entrada (em Hertz). Amazon Transcribe suporta uma faixa de 8.000 Hz a 48.000 Hz. O áudio de baixa qualidade, como o áudio do telefone, normalmente está em torno de 8.000 Hz. O áudio de alta qualidade geralmente varia de 16.000 Hz a 48.000 Hz. Observe que a taxa de amostragem especificada deve corresponder à do seu áudio.
- `x-amzn-transcribe-vocabulary-name`: O nome do vocabulário que você deseja usar com sua transcrição.
- `x-amzn-transcribe-specialty`: A especialidade médica que está sendo transcrita.
- `x-amzn-transcribe-type`: Escolha se isso é um ditado ou uma conversa.
- `x-amzn-transcribe-show-speaker-label`: para ativar a diarização, esse valor deve ser `true`.
- tipo de conteúdo: não altere esse campo; use o conteúdo mostrado no exemplo anterior.

Habilitando o particionamento de alto-falantes em uma WebSocket solicitação

Para particionar alto-falantes em WebSocket streams com a API, use o formato a seguir para criar um URI pré-assinado para iniciar uma WebSocket solicitação e definir `vocabulary-name` o nome do vocabulário personalizado.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket
?language-code=en-US
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&media-encoding=flac
&sample-rate=16000
&session-id=sessionId
&specialty=medicalSpecialty
```

```
&type=CONVERSATION
&vocabulary-name=vocabularyName
&show-speaker-label=boolean
```

Conjunto de caracteres para Amazon Transcribe Medicina

Para usar vocabulários personalizados em Amazon Transcribe Medicina, use o seguinte conjunto de caracteres.

Conjunto de caracteres em inglês

Para vocabulários personalizados em inglês, você pode usar os seguintes caracteres nas colunas Phrase e SoundsLike:

- a - z
- A - Z
- ' (apóstrofo)
- - (hífen)
- . (ponto final)

Você pode usar os caracteres do Alfabeto Fonético Internacional (IPA) a seguir no campo IPA do arquivo de entrada de vocabulário:

Caractere	Código	Caractere	Código
aʊ	0061 028A	w	0077
aɪ	0061 026A	z	007A
b	0062	æ	00E6
d	0064	ð	00F0
eɪ	0065 026A	ŋ	014B
f	0066	ɑ	0251
g	0067	ɔ	0254

Caractere	Código	Caractere	Código
h	0068	ɔɪ	0254 026A
i	0069	ə	0259
j	006A	ɛ	025B
k	006B	ʒ	025D
l	006C	g	0261
ɹ	006C 0329	ɪ	026A
m	006D	ɹ	0279
n	006E	ʃ	0283
ŋ	006E 0329	ʊ	028A
oʊ	006F 028A	ʌ	028C
p	0070	ɹ	028D
s	0073	ʒ	0292
t	0074	ɔʒ	02A4
u	0075	tʃ	02A7
v	0076	θ	03B8

Identificando informações de saúde pessoal (PHI) em uma transcrição

Use a Identificação de Informações Pessoais de Health para rotular informações pessoais de saúde (PHI) nos resultados da transcrição. Ao revisar os rótulos, você pode encontrar PHI que podem ser usados para identificar um paciente.

Você pode identificar PHI usando um stream em tempo real ou um trabalho de transcrição em lote.

Você pode usar seu próprio pós-processamento para redigir o PHI identificado na saída da transcrição.

Use a Identificação de Informações Pessoais de Health para identificar os seguintes tipos de PHI:

- PHI pessoal:
 - Nomes — Nome completo ou sobrenome e inicial
 - Sexo
 - Idade
 - Números de telefone
 - Datas (não incluindo o ano) que se relacionam diretamente com o paciente
 - Endereços de e-mail
- PHI geográfico:
 - Endereço físico
 - Código postal
 - Nome do centro médico ou consultório
- PHI da conta:
 - Números de fax
 - Números de previdência social (SSNs)
 - Números de beneficiários do seguro Health
 - AWS Account Numbers
 - Certificado ou números de licença
- PHI do veículo:
 - Número de identificação de veículo (VIN)
 - Número da placa
- Outras PHI:
 - Localização uniforme de recursos na Web (URL)
 - Números de endereço IP (Internet Protocol)

Amazon TranscribeO Medical é um serviço elegível pela Lei de Portabilidade e Responsabilidade de Provedores de Health de 1996 (HIPAA) dos EUA Para obter mais informações, consulte [Amazon Transcribe Medical](#). Para obter informações sobre como identificar PHI em um arquivo de áudio,

consulte [Identificando PHI em um arquivo de áudio](#). Para obter informações sobre como identificar PHI em um stream, consulte [Identificando PHI em um stream em tempo real](#).

Tópicos

- [Identificando PHI em um arquivo de áudio](#)
- [Identificando PHI em um stream em tempo real](#)

Identificando PHI em um arquivo de áudio

Use um trabalho de transcrição em lote para transcrever arquivos de áudio e identificar as informações pessoais de saúde (PHI) dentro deles. Quando você ativa a Identificação de Informações Pessoais de Health (PHI), a Amazon Transcribe Medical rotula a PHI que identificou nos resultados da transcrição. Para obter informações sobre o PHI que o Amazon Transcribe Medical pode identificar, consulte [Identificando informações de saúde pessoal \(PHI\) em uma transcrição](#).

Você pode iniciar um trabalho de transcrição em lote usando a [StartMedicalTranscriptionJob](#) API ou AWS Management Console o.

AWS Management Console

Para usar o diálogo AWS Management Console para transcrever um diálogo médico-paciente, crie um trabalho de transcrição e escolha Conversa para o tipo de entrada de áudio.

Para transcrever um arquivo de áudio e identificar seu PHI (AWS Management Console)

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, em Amazon Transcribe Médico, escolha Provedores de transcrição.
3. Escolha Create job (Criar trabalho).
4. Na página Especificar detalhes do Job, em Configurações do trabalho, especifique o seguinte.
 - a. Nome — O nome do trabalho de transcrição que é exclusivo para sua Conta da AWS.
 - b. Tipo de entrada de áudio — Conversa ou ditado.
5. Para os campos restantes, especifique a Amazon S3 localização do seu arquivo de áudio e onde você deseja armazenar a saída do trabalho de transcrição.
6. Escolha Próximo.
7. Em Configurações de áudio, escolha Identificação PHI.
8. Escolha Create (Criar).

API

Para transcrever um arquivo de áudio e identificar seu PHI usando um trabalho de transcrição em lote (API)

- Para a [StartMedicalTranscriptionJob](#) API, especifique o seguinte.
 - a. Para `MedicalTranscriptionJobName`, especifique um nome exclusivo para sua Conta da AWS.
 - b. Para `LanguageCode`, especifique o código do idioma que corresponde ao idioma falado no seu arquivo de áudio.
 - c. Para o `MediaFileUri` parâmetro do `Media` objeto, especifique o nome do arquivo de áudio que você deseja transcrever.
 - d. Para `Specialty`, especifique a especialidade médica do médico que fala no arquivo de áudio como `PRIMARYCARE`.
 - e. Para o `Type`, especifique `CONVERSATION` ou `DICTATION`.
 - f. Para `OutputBucketName`, especifique o Amazon S3 bucket no qual deseja armazenar os resultados da transcrição.

A seguir está um exemplo de solicitação que usa o AWS SDK for Python (Boto3) para transcrever um arquivo de áudio e identificar o PHI de um paciente.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe')
job_name = "my-first-transcription-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac"
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {'MediaFileUri': job_uri},
    LanguageCode = 'en-US',
    ContentIdentificationType = 'PHI',
    Specialty = 'PRIMARYCARE',
    Type = 'type', # Specify 'CONVERSATION' for a medical conversation. Specify
    'DICTATION' for a medical dictation.
    OutputBucketName = 'DOC-EXAMPLE-BUCKET'
)
```

```
while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
    job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
    'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

O código de exemplo a seguir mostra os resultados da transcrição com o PHI do paciente identificado.

```
{
  "jobName": "my-medical-transcription-job-name",
  "accountId": "111122223333",
  "results": {
    "transcripts": [{
      "transcript": "The patient's name is Bertrand."
    }],
    "items": [{
      "start_time": "0.0",
      "end_time": "0.37",
      "alternatives": [{
        "confidence": "0.9993",
        "content": "The"
      }],
      "type": "pronunciation"
    }, {
      "start_time": "0.37",
      "end_time": "0.44",
      "alternatives": [{
        "confidence": "0.9981",
        "content": "patient's"
      }],
      "type": "pronunciation"
    }, {
      "start_time": "0.44",
      "end_time": "0.52",
      "alternatives": [{
```

```
        "confidence": "1.0",
        "content": "name"
    }],
    "type": "pronunciation"
}, {
    "start_time": "0.52",
    "end_time": "0.92",
    "alternatives": [{
        "confidence": "1.0",
        "content": "is"
    }],
    "type": "pronunciation"
}, {
    "start_time": "0.92",
    "end_time": "0.9989",
    "alternatives": [{
        "confidence": "1.0",
        "content": "Bertrand"
    }],
    "type": "pronunciation"
}, {
    "alternatives": [{
        "confidence": "0.0",
        "content": "."
    }],
    "type": "punctuation"
}],
"entities": [{
    "content": "Bertrand",
    "category": "PHI*-Personal*",
    "startTime": 0.92,
    "endTime": 1.2,
    "confidence": 0.9989
}],
},
"status": "COMPLETED"
}
```

AWS CLI

Para transcrever um arquivo de áudio e identificar PHI usando um trabalho de transcrição em lote (AWS CLI)

- Execute o código a seguir.

```
aws transcribe start-medical-transcription-job \  
--medical-transcription-job-name my-medical-transcription-job-name \  
--language-code en-US \  
--media MediaFileUri="s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac" \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--specialty PRIMARYCARE \  
--type type \ # Choose CONVERSATION to transcribe a medical conversation.  
Choose DICTATION to transcribe a medical dictation.  
--content-identification-type PHI
```

Identificando PHI em um stream em tempo real

Você pode identificar informações pessoais de Health (PHI) em HTTP/2 ou WebSocket streams. Quando você ativa a Identificação PHI, a Amazon Transcribe Medical rotula a PHI que identifica nos resultados da transcrição. Para obter informações sobre o PHI que o Amazon Transcribe Medical pode identificar, consulte [Identificando informações de saúde pessoal \(PHI\) em uma transcrição](#).

Identificando PHI em um ditado falado em seu microfone

Para usar o AWS Management Console para transcrever a fala captada pelo microfone e identificar qualquer PHI, escolha Ditado como o tipo de entrada de áudio, inicie a transmissão e comece a falar no microfone do seu computador.

Para identificar PHI em um ditado usando o AWS Management Console

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, escolha Transcrição em tempo real.
3. Para Tipo de entrada de áudio, escolha Ditado.
4. Para Configurações adicionais, escolha Identificação PHI.

5. Escolha Iniciar streaming e fale no microfone.
6. Escolha Interromper streaming para encerrar o ditado.

Identificando PHI em um stream HTTP/2

Para iniciar um stream HTTP/2 com a identificação PHI ativada, use a [StartMedicalStreamTranscription](#) API e especifique o seguinte:

- Para `LanguageCode`, especifique o código do idioma falado no stream. Para inglês dos EUA, especifique `en-US`.
- Para `MediaSampleHertz`, especifique a taxa de amostragem do áudio.
- Para `content-identification-type`, especifique PHI.

Identificando PHI em um WebSocket stream

Para iniciar um WebSocket stream com a identificação PHI ativada, use o formato a seguir para criar uma URL pré-assinada.

```
GET wss://transcribestreaming.us-west-2.amazonaws.com:8443/medical-stream-
transcription-websocket?
&X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20220208%2Fus-
west-2%2Ftranscribe%2Faws4_request
&X-Amz-Date=20220208T235959Z
&X-Amz-Expires=300
&X-Amz-Security-Token=security-token
&X-Amz-Signature=Signature Version 4 signature
&X-Amz-SignedHeaders=host
&language-code=en-US
&media-encoding=flac
&sample-rate=16000
&specialty=medical-specialty
&content-identification-type=PHI
```

As definições de parâmetros podem ser encontradas na [Referência da API](#); os parâmetros comuns a todas as operações AWS da API estão listados na seção [Parâmetros comuns](#).

Gerando transcrições alternativas

Quando você usa o Amazon Transcribe Medical, você obtém a transcrição com o nível de confiança mais alto. No entanto, você pode configurar o Amazon Transcribe Medical para retornar transcrições adicionais com níveis de confiança mais baixos.

Use transcrições alternativas para ver diferentes interpretações do áudio transcrito. Por exemplo, em um aplicativo que permite que uma pessoa revise a transcrição, é possível apresentar as transcrições alternativas para a pessoa a escolher.

Você pode gerar transcrições alternativas com o AWS Management Console ou a [StartMedicalTranscriptionJobAPI](#).

AWS Management Console

Para usar o AWS Management Console para gerar transcrições alternativas, você ativa resultados alternativos ao configurar seu trabalho.

1. Faça login no [AWS Management Console](#).
2. No painel de navegação, em Amazon Transcribe Médica, escolha Tarefas de transcrição.
3. Escolha Create job (Criar trabalho).
4. Na página Especificar detalhes do trabalho, forneça informações sobre seu trabalho de transcrição.
5. Escolha Next (Próximo).
6. Habilite resultados alternativos.
7. Em Alternativas máximas, insira um valor inteiro entre 2 e 10, para o número máximo de transcrições alternativas que você deseja na saída.
8. Escolha Create (Criar).

API

Para separar texto por alto-falante em um arquivo de áudio usando um trabalho de transcrição em lote (API)

- Para a [StartMedicalTranscriptionJobAPI](#), especifique o seguinte.

- a. Para `MedicalTranscriptionJobName`, especifique um nome exclusivo em sua Conta da AWS.
- b. Para `LanguageCode`, especifique o código do idioma que corresponde ao idioma falado no seu arquivo de áudio e ao idioma do seu filtro de vocabulário.
- c. Para o parâmetro `MediaFileUri` do objeto `Media`, especifique a localização do arquivo de áudio que você deseja transcrever.
- d. Para `Specialty`, especifique a especialidade médica do médico que fala no arquivo de áudio.
- e. Para `Type`, especifique se você está transcrevendo uma conversa médica ou um ditado.
- f. Para `OutputBucketName`, especifique o Amazon S3 bucket para armazenar os resultados da transcrição.
- g. Para o objeto `Settings`, especifique o seguinte.
 - i. `ShowAlternatives` – `true`.
 - ii. `MaxAlternatives` – Um número inteiro entre 2 e 10 para indicar o número de transcrições alternativas que você deseja na saída da transcrição.

A solicitação a seguir usa o AWS SDK for Python (Boto3) para iniciar um trabalho de transcrição que gera até duas transcrições alternativas.

```
from __future__ import print_function
import time
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-transcription-job"
job_uri = s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-file.flac
transcribe.start_medical_transcription_job(
    MedicalTranscriptionJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    OutputKey = 'my-output-files/',
    LanguageCode = 'en-US',
    Specialty = 'PRIMARYCARE',
    Type = 'CONVERSATION',
```



```
    Settings = {
        'ShowAlternatives': True,
        'MaxAlternatives': 2
    }
)

while True:
    status = transcribe.get_medical_transcription_job(MedicalTranscriptionJobName =
job_name)
    if status['MedicalTranscriptionJob']['TranscriptionJobStatus'] in ['COMPLETED',
'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

AWS CLI

Para transcrever um arquivo de áudio de uma conversa entre um médico de cuidados primários e um paciente em um arquivo de áudio (AWS CLI)

- Execute o código a seguir.

```
aws transcribe start-transcription-job \  
--cli-input-json file://filepath/example-start-command.json
```

O Pares mostra o conteúdo de `example-start-command.json`.

```
{
  "MedicalTranscriptionJobName": "my-first-transcription-job",
  "LanguageCode": "en-US",
  "Specialty": "PRIMARYCARE",
  "Type": "CONVERSATION",
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-audio-  
file.flac"
  }
}
```

```
    },  
    "Settings":{  
        "ShowAlternatives": true,  
        "MaxAlternatives": 2  
    }  
}
```

Amazon Transcribe Endpoints de VPC médicos e de interface (AWS PrivateLink)

Você pode estabelecer uma conexão privada entre sua VPC e Amazon Transcribe Médico, criando um terminal VPC de interface. Os endpoints de interface são alimentados por [AWS PrivateLink](#), uma tecnologia que permite que você acesse de forma privada Amazon Transcribe APIs médicas sem gateway de internet, dispositivo NAT, conexão VPN ou AWS Direct Connect conexão. As instâncias em sua VPC não precisam de endereços IP públicos para se comunicar Amazon Transcribe APIs médicas. Tráfego entre sua VPC e Amazon Transcribe A Medical não sai da rede Amazon.

Cada endpoint de interface é representado por uma ou mais [interfaces de rede elástica](#) nas sub-redes.

Para obter mais informações, consulte [Endpoints VPC de interface \(AWS PrivateLink\)](#) na Amazon VPC Guia do usuário.

Considerações para Amazon Transcribe Endpoints médicos de VPC

Antes de configurar um endpoint VPC de interface para Amazon Transcribe Médico, certifique-se de revisar [Propriedades e limitações do endpoint da interface](#) na Amazon VPC Guia do usuário.

Amazon Transcribe O Medical oferece suporte para fazer chamadas para todas as suas ações de API a partir do seu VPC.

Criação de um endpoint VPC de interface para Amazon Transcribe Médico

Você pode criar um endpoint VPC para o Amazon Transcribe Serviço médico usando o AWS Management Console ou o AWS CLI. Para obter mais informações, consulte [Criar um endpoint de interface](#) no Guia do usuário da Amazon VPC.

Para transcrição em lote em Amazon Transcribe Médico, crie um endpoint VPC usando o seguinte nome de serviço:

- `com.amazonaws.us-west-2.transcrever`

Para streaming de transcrição em Amazon Transcribe Médico, crie um endpoint VPC usando o seguinte nome de serviço:

- `com.amazonaws.us-west-2.transcrever-streaming`

Se você habilitar o DNS privado para o endpoint, poderá fazer solicitações de API para Amazon Transcribe Médico usando seu nome DNS padrão para a Região da AWS, por exemplo, `transcribestreaming.us-east-2.amazonaws.com`.

Para obter mais informações, consulte [Acessando um serviço por meio de um endpoint de interface na Amazon VPC](#) Guia do usuário.

Criação de uma política de endpoint de VPC para Amazon Transcribe Streaming médico

Você pode anexar uma política de endpoint ao seu endpoint VPC que controla o acesso a Amazon Transcribe Médico. Essa política especifica as seguintes informações:

- A entidade principal que pode executar ações.
- As ações que podem ser executadas.
- Os recursos sobre os quais as ações podem ser realizadas.

Para obter mais informações, consulte [Controlar o acesso a serviços com VPC endpoints](#) no Guia do usuário da Amazon VPC.

Exemplo: política de endpoint de VPC para Amazon Transcribe Ações de transcrição de streaming médico

Veja a seguir um exemplo de uma política de endpoint para transcrição de streaming em Amazon Transcribe Médico. Quando anexada a um endpoint, essa política concede acesso ao listado Amazon Transcribe Ações médicas para todos os diretores em todos os recursos.

```
{
  "Statement": [
    {
      "Principal": "*",
```

```

    "Effect": "Allow",
    "Action": [
        "transcribe:StartMedicalStreamTranscription",
    ],
    "Resource": "*"
  }
]
}

```

Exemplo: política de endpoint de VPC para Amazon Transcribe Ações médicas de transcrição em lote

Veja a seguir um exemplo de uma política de endpoint para transcrição em lote em Amazon Transcribe Médico. Quando anexada a um endpoint, essa política concede acesso ao listado Amazon Transcribe Ações médicas para todos os diretores em todos os recursos.

```

{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "transcribe:StartMedicalTranscriptionJob"
      ],
      "Resource": "*"
    }
  ]
}

```

Sub-redes compartilhadas

Você não pode criar, descrever, modificar ou excluir endpoints VPC em sub-redes que são compartilhadas com você. No entanto, você pode usar os endpoints da VPC em sub-redes que são compartilhadas com você. Para obter informações sobre o compartilhamento de VPC, consulte [Compartilhe sua VPC com outras contas](#) na Amazon Virtual Private Cloud guia.

AWS HealthScribe

O AWS HealthScribe é um novo recurso de machine learning (ML) qualificado pela HIPAA que combina reconhecimento de fala e IA generativa para transcrever conversas entre pacientes e médicos e gerar notas clínicas fáceis de revisar. O AWS HealthScribe ajuda os fornecedores de software de saúde a criar aplicações clínicas que reduzem a carga de documentação e melhoram a experiência de consulta. O serviço fornece automaticamente transcrições avançadas de conversas, identifica as funções do palestrante, classifica diálogos, extrai termos médicos e gera notas clínicas preliminares. O AWS HealthScribe combina esses recursos para eliminar a necessidade de integrar e otimizar serviços de IA separados, permitindo que você agilize a implementação.

Casos de uso comuns:

- Reduzir o tempo de documentação: permita que os médicos concluam rapidamente a documentação clínica com notas clínicas geradas por IA que são fáceis de revisar, ajustar e finalizar em sua aplicação.
- Aumentar a eficiência do escrevente médico: equipe os escreventes médicos com notas clínicas e transcrições geradas por IA, juntamente com o áudio da consulta, para agilizar o tempo de entrega da documentação.
- Resumo eficiente da visita ao paciente: crie uma experiência que permita aos usuários lembrar rapidamente os principais destaques de suas conversas na aplicação.

Important

Os resultados produzidos pelo AWS HealthScribe são probabilísticos e nem sempre são precisos devido a vários fatores, incluindo qualidade de áudio, ruído de fundo, clareza de quem fala, complexidade da terminologia médica, nuances linguísticas específicas do contexto e a [natureza do machine learning e da IA generativa](#). O AWS HealthScribe foi desenvolvido para ser usado como auxiliar por médicos e escreventes médicos. A saída do HealthScribe só deve ser usada em cenários de atendimento ao paciente, incluindo, mas não de forma exclusiva, como parte dos prontuários eletrônicos, após análise da precisão e imposição de um bom julgamento médico por profissionais médicos treinados. A saída do HealthScribe não substitui o aconselhamento, diagnóstico ou tratamento médico profissional e não se destina a curar, tratar, mitigar, prevenir ou diagnosticar qualquer doença ou condição de saúde.

O AWS HealthScribe opera sob um modelo de responsabilidade compartilhada, no qual a AWS é responsável por proteger a infraestrutura que executa o AWS HealthScribe e você é responsável por gerenciar seus dados. Para obter mais informações, consulte o [Modelo de responsabilidade compartilhada](#).

O AWS HealthScribe está disponível na região Leste dos EUA (Norte da Virgínia).

O serviço está disponível em inglês dos EUA (en-US). Para conseguir melhores resultados, use um formato de áudio sem perdas, como FLAC ou WAV, com codificação PCM de 16 bits. O AWS HealthScribe permite taxas de amostragem de 16.000 Hz ou superior.

Atualmente, o AWS HealthScribe é compatível com as especialidades de Medicina Geral e Ortopedia.

Um trabalho do AWS HealthScribe analisa a consulta médica para produzir dois arquivos de saída JSON: um arquivo de [transcrição](#) e um arquivo de [documentação clínica](#).

No arquivo de transcrição, além da saída padrão de transcrição passo a passo com registros de data e hora no nível da palavra, o AWS HealthScribe fornece:

- Detecção da função do participante para que você possa distinguir os pacientes dos médicos na transcrição da conversa.
- Seção de transcrição, que categoriza os diálogos da transcrição com base em sua relevância clínica, como conversa trivial, subjetiva, objetiva etc. Isso pode ser usado para mostrar partes específicas da transcrição.
- Entidades clínicas, que incluem informações estruturadas, como medicamentos, condições médicas e tratamentos mencionados na conversa.

No arquivo da documentação clínica, o AWS HealthScribe fornece:

- Resumos contendo notas resumidas das principais seções da documentação clínica, como queixa principal, histórico da doença atual, revisão de sistemas, histórico médico anterior, avaliação e plano.
- Links de evidências que vinculam cada frase usada nos resumos gerados pela IA à transcrição original da consulta, facilitando aos usuários a validação da precisão do resumo na aplicação.

Operações de API específicas do AWS HealthScribe:

- StartMedicalScribeJob
- ListMedicalScribeJobs
- GetMedicalScribeJob
- DeleteMedicalScribeJob

Consulte exemplos de solicitações do AWS HealthScribe em [Starting an AWS HealthScribe job](#).

Arquivo de transcrição

O arquivo de transcrição fornece o conteúdo da conversa em um formato passo a passo.

Além disso, os seguintes insights são fornecidos para cada turno de conversa:

- Função do participante: cada participante é rotulado como médico ou paciente. Se uma conversa tiver mais de um participante em cada categoria, cada participante receberá um número. Por exemplo, CLINICIAN_1, CLINICIAN_2 e PATIENT_1, PATIENT_2.
- Seção: cada turno do diálogo é atribuído a uma das cinco seções possíveis com base no conteúdo identificado.
 - Subjetivo: informações fornecidas pelo paciente sobre seus problemas de saúde.
 - Objetivo: informações observadas pelo médico por meio de exames físicos, laboratoriais, de imagem ou testes de diagnóstico.
 - Avaliação e plano: informações relacionadas à avaliação do médico e ao plano de tratamento.
 - Gerenciamento de fluxo da consulta: informações relacionadas a conversas triviais ou transições.
- Insights: extrai entidades clinicamente relevantes (ClinicalEntity) presentes na conversa. AWS O HealthScribe detecta todas as entidades clínicas compatíveis com o [Amazon Comprehend Medical](#).

Consulte informações de saída mais detalhadas em [Example transcript output](#).

Arquivo de documentação clínica

O arquivo de informações da documentação contém resumos das seguintes seções principais da documentação clínica.

Seção	Descrição
QUEIXA PRINCIPAL	Breve descrição do motivo pelo qual o paciente visitou o médico.
HISTÓRIA DA DOENÇA ATUAL	Notas que fornecem informações sobre a doença do paciente, incluindo referência à gravidade, início, tempo dos sintomas, tratamentos atuais e áreas afetadas.
REVISÃO DE SISTEMAS	Avaliação dos sintomas relatada pelo paciente em diferentes sistemas corporais.
HISTÓRICO MÉDICO	Detalha as condições médicas, cirurgias e tratamentos anteriores de um paciente.
AVALIAÇÃO	Notas que fornecem informações sobre a avaliação do médico sobre a saúde do paciente.
PLAN	Notas que fazem referência a tratamentos médicos, ajustes no estilo de vida e consultas adicionais.

Cada frase presente no Summary inclui referências à transcrição original da consulta, facilitando aos usuários validar a precisão do resumo na aplicação. Fornecer rastreabilidade e transparência para insights gerados pela IA é consistente com os princípios da IA responsável, como explicabilidade. Fornecer essas referências junto com as notas de resumo aos médicos ou escreventes médicos ajuda a promover a confiança e incentivar o uso seguro da IA nos ambientes clínicos.

Cada frase no Summary vem com EvidenceLinks que fornecem o SegmentId dos diálogos relevantes na transcrição que foram resumidos.

Consulte informações de saída mais detalhadas em [Example clinical documentation output](#).

Iniciar um trabalho do AWS HealthScribe

É possível iniciar um trabalho do AWS HealthScribe usando a AWS CLI ou os AWS SDKs; veja exemplos a seguir.

AWS CLI

Este exemplo usa o comando [start-medical-scribe-job](#). Consulte mais informações em [StartMedicalScribeJob](#).

```
aws transcribe start-medical-scribe-job \  
--region us-west-2 \  
--medical-scribe-job-name my-first-medical-scribe-job \  
--media MediaFileUri=s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac \  
--output-bucket-name DOC-EXAMPLE-BUCKET \  
--DataAccessRoleArn=arn:aws:iam::111122223333:role/ExampleRole \  
--settings ShowSpeakerLabels=false,ChannelIdentification=true \  
--channel-definitions ChannelId=0,ParticipantRole=CLINICIAN \  
ChannelId=1,ParticipantRole=PATIENT
```

Veja a seguir outro exemplo usando o comando [start-medical-scribe-job](#) e um corpo de solicitação com configurações adicionais.

```
aws transcribe start-medical-scribe-job \  
--region us-west-2 \  
--cli-input-json file://filepath/my-first-medical-scribe-job.json
```

O arquivo `my-first-medical-scribe-job.json` contém o seguinte corpo de solicitação.

```
{  
  "MedicalScribeJobName": "my-first-medical-scribe-job",  
  "Media": {  
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"  
  },  
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",  
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole",  
  "Settings": {
```

```

    "ShowSpeakerLabels": false,
    "ChannelIdentification": true
  },
  "ChannelDefinitions": [
    {
      "ChannelId": 0,
      "ParticipantRole": "CLINICIAN"
    }, {
      "ChannelId": 1,
      "ParticipantRole": "PATIENT"
    }
  ]
}

```

AWS SDK for Python (Boto3)

O exemplo a seguir usa o AWS SDK for Python (Boto3) para fazer uma solicitação [start_medical_scribe_job](#). Consulte mais informações em [StartMedicalScribeJob](#).

```

from __future__ import print_function
import boto3
transcribe = boto3.client('transcribe', 'us-west-2')
job_name = "my-first-medical-scribe-job"
job_uri = "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
transcribe.start_medical_scribe_job(
    MedicalScribeJobName = job_name,
    Media = {
        'MediaFileUri': job_uri
    },
    OutputBucketName = 'DOC-EXAMPLE-BUCKET',
    DataAccessRoleArn = 'arn:aws:iam::111122223333:role/ExampleRole',
    Settings = {
        'ShowSpeakerLabels': false,
        'ChannelIdentification': true
    },
    ChannelDefinitions = [
        {
            'ChannelId': 0,
            'ParticipantRole': 'CLINICIAN'
        }, {
            'ChannelId': 1,
            'ParticipantRole': 'PATIENT'
        }
    ]
)

```

```
    }
  ]
)
while True:
    status = transcribe.get_medical_scribe_job(MedicalScribeJobName = job_name)
    if status['MedicalScribeJob']['MedicalScribeJobStatus'] in ['COMPLETED', 'FAILED']:
        break
    print("Not ready yet...")
    time.sleep(5)
print(status)
```

Note

No momento, o Console de Gerenciamento da AWS não é compatível com trabalhos do AWS HealthScribe.

Exemplo de saída

Além de uma transcrição, as solicitações `StartMedicalScribeJob` geram um arquivo de documentação clínica separado. Ambos os arquivos estão no formato JSON e são armazenados no local de saída especificado na solicitação. Veja alguns exemplos de cada tipo de saída:

Exemplo de saída de transcrição

Um arquivo de transcrição do AWS HealthScribe (de uma solicitação `StartMedicalScribeJob`) tem o seguinte formato:

```
{
  "Conversation": {
    "ConversationId": "sampleConversationUUID",
    "JobName": "sampleJobName",
    "JobType": "ASYNC",
    "LanguageCode": "en-US",
    "ClinicalInsights": [
      {
        "Attributes": [],
        "Category": "MEDICAL_CONDITION",
        "InsightId": "insightUUID1",
        "InsightType": "ClinicalEntity",
```

```
"Spans": [
  {
    "BeginCharacterOffset": 12,
    "Content": "pain",
    "EndCharacterOffset": 15,
    "SegmentId": "uuid1"
  }
],
>Type": "DX_NAME"
},
{
  "Attributes": [],
  "Category": "TEST_TREATMENT_PROCEDURE",
  "InsightId": "insightUUID2",
  "InsightType": "ClinicalEntity",
  "Spans": [
    {
      "BeginCharacterOffset": 4,
      "Content": "mammogram",
      "EndCharacterOffset": 12,
      "SegmentId": "uuid2"
    }
  ],
>Type": "TEST_NAME"
},
{
  "Attributes": [],
  "Category": "TEST_TREATMENT_PROCEDURE",
  "InsightId": "insightUUID3",
  "InsightType": "ClinicalEntity",
  "Spans": [
    {
      "BeginCharacterOffset": 15,
      "Content": "pap smear",
      "EndCharacterOffset": 23,
      "SegmentId": "uuid3"
    }
  ],
>Type": "TEST_NAME"
},
{
  "Attributes": [],
  "Category": "MEDICATION",
  "InsightId": "insightUUID4",
```

```
"InsightType": "ClinicalEntity",
"Spans": [
  {
    "BeginCharacterOffset": 28,
    "Content": "phentermine",
    "EndCharacterOffset": 38,
    "SegmentId": "uuid4"
  }
],
>Type": "GENERIC_NAME"
},
{
  "Attributes": [
    {
      "AttributeId": "attributeUUID1",
      "Spans": [
        {
          "BeginCharacterOffset": 38,
          "Content": "high",
          "EndCharacterOffset": 41,
          "SegmentId": "uuid5"
        }
      ],
      "Type": "TEST_VALUE"
    }
  ],
  "Category": "TEST_TREATMENT_PROCEDURE",
  "InsightId": "insightUUID5",
  "InsightType": "ClinicalEntity",
  "Spans": [
    {
      "BeginCharacterOffset": 14,
      "Content": "weight",
      "EndCharacterOffset": 19,
      "SegmentId": "uuid6"
    }
  ],
  "Type": "TEST_NAME"
},
{
  "Attributes": [],
  "Category": "ANATOMY",
  "InsightId": "insightUUID6",
  "InsightType": "ClinicalEntity",
```

```
    "Spans": [
      {
        "BeginCharacterOffset": 60,
        "Content": "heart",
        "EndCharacterOffset": 64,
        "SegmentId": "uuid7"
      }
    ],
    "Type": "SYSTEM_ORGAN_SITE"
  }
],
"TranscriptItems": [
  {
    "Alternatives": [
      {
        "Confidence": 0.7925,
        "Content": "Okay"
      }
    ],
    "BeginAudioTime": 0.16,
    "EndAudioTime": 0.6,
    "Type": "PRONUNCIATION"
  },
  {
    "Alternatives": [
      {
        "Confidence": 0,
        "Content": "."
      }
    ],
    "BeginAudioTime": 0,
    "EndAudioTime": 0,
    "Type": "PUNCTUATION"
  },
  {
    "Alternatives": [
      {
        "Confidence": 1,
        "Content": "Good"
      }
    ],
    "BeginAudioTime": 0.61,
    "EndAudioTime": 0.92,
    "Type": "PRONUNCIATION"
  }
]
```

```
},
{
  "Alternatives": [
    {
      "Confidence": 1,
      "Content": "afternoon"
    }
  ],
  "BeginAudioTime": 0.92,
  "EndAudioTime": 1.54,
  "Type": "PRONUNCIATION"
},
{
  "Alternatives": [
    {
      "Confidence": 0,
      "Content": "."
    }
  ],
  "BeginAudioTime": 0,
  "EndAudioTime": 0,
  "Type": "PUNCTUATION"
},
{
  "Alternatives": [
    {
      "Confidence": 0.9924,
      "Content": "You"
    }
  ],
  "BeginAudioTime": 1.55,
  "EndAudioTime": 1.88,
  "Type": "PRONUNCIATION"
},
{
  "Alternatives": [
    {
      "Confidence": 1,
      "Content": "lost"
    }
  ],
  "BeginAudioTime": 1.88,
  "EndAudioTime": 2.19,
  "Type": "PRONUNCIATION"
}
```

```
    },
    {
      "Alternatives": [
        {
          "Confidence": 1,
          "Content": "one"
        }
      ],
      "BeginAudioTime": 2.19,
      "EndAudioTime": 2.4,
      "Type": "PRONUNCIATION"
    },
    {
      "Alternatives": [
        {
          "Confidence": 1,
          "Content": "lb"
        }
      ],
      "BeginAudioTime": 2.4,
      "EndAudioTime": 2.97,
      "Type": "PRONUNCIATION"
    }
  ],
  "TranscriptSegments": [
    {
      "BeginAudioTime": 0.16,
      "Content": "Okay.",
      "EndAudioTime": 0.6,
      "ParticipantDetails": {
        "ParticipantRole": "CLINICIAN_0"
      },
      "SectionDetails": {
        "SectionName": "SUBJECTIVE"
      },
      "SegmentId": "uuid1"
    },
    {
      "BeginAudioTime": 0.61,
      "Content": "Good afternoon.",
      "EndAudioTime": 1.54,
      "ParticipantDetails": {
        "ParticipantRole": "CLINICIAN_0"
      }
    }
  ],
```



```
"SectionDetails": {
  "SectionName": "OTHER"
},
"SegmentId": "uuid2"
},
{
  "BeginAudioTime": 1.55,
  "Content": "You lost one lb.",
  "EndAudioTime": 2.97,
  "ParticipantDetails": {
    "ParticipantRole": "CLINICIAN_0"
  },
  "SectionDetails": {
    "SectionName": "SUBJECTIVE"
  },
  "SegmentId": "uuid3"
},
{
  "BeginAudioTime": 2.98,
  "Content": "Yeah, I think it, uh, do you feel more energy?",
  "EndAudioTime": 6.95,
  "ParticipantDetails": {
    "ParticipantRole": "CLINICIAN_0"
  },
  "SectionDetails": {
    "SectionName": "SUBJECTIVE"
  },
  "SegmentId": "uuid5"
},
{
  "BeginAudioTime": 6.96,
  "Content": "Yes.",
  "EndAudioTime": 7.88,
  "ParticipantDetails": {
    "ParticipantRole": "CLINICIAN_0"
  },
  "SectionDetails": {
    "SectionName": "SUBJECTIVE"
  },
  "SegmentId": "uuid6"
},
{
  "BeginAudioTime": 7.89,
```

```

    "Content": "Uh, how about craving for the carbohydrate or sugar or fat or
anything?",
    "EndAudioTime": 17.93,
    "ParticipantDetails": {
      "ParticipantRole": "CLINICIAN_0"
    },
    "SectionDetails": {
      "SectionName": "SUBJECTIVE"
    },
    "SegmentId": "uuid7"
  }
]
}
}

```

Veja a seguir outro exemplo usando o comando [start-medical-scribe-job](#) e um corpo de solicitação com configurações adicionais.

```

aws transcribe start-medical-scribe-job \
--region us-west-2 \
--cli-input-json file://filepath/my-first-medical-scribe-job.json

```

O arquivo `my-first-medical-scribe-job.json` contém o seguinte corpo de solicitação.

```

{
  "MedicalScribeJobName": "my-first-medical-scribe-job",
  "Media": {
    "MediaFileUri": "s3://DOC-EXAMPLE-BUCKET/my-input-files/my-media-file.flac"
  },
  "OutputBucketName": "DOC-EXAMPLE-BUCKET",
  "DataAccessRoleArn": "arn:aws:iam::111122223333:role/ExampleRole",
  "Settings": {
    "ShowSpeakerLabels": false,
    "ChannelIdentification": true
  },
  "ChannelDefinitions": [
    {
      "ChannelId": 0,
      "ParticipantRole": "CLINICIAN"
    }
  ]
}

```

```
    }, {
      "ChannelId": 1,
      "ParticipantRole": "PATIENT"
    }
  ]
}
```

Exemplo de saída de documentação clínica

Um arquivo de insights de documentação (de uma solicitação `StartMedicalScribeJob`) tem o seguinte formato:

```
{
  "ClinicalDocumentation": {
    "Sections": [
      {
        "SectionName": "CHIEF_COMPLAINT",
        "Summary": [
          {
            "EvidenceLinks": [
              {
                "SegmentId": "uuid1"
              },
              {
                "SegmentId": "uuid2"
              },
              {
                "SegmentId": "uuid3"
              },
              {
                "SegmentId": "uuid4"
              },
              {
                "SegmentId": "uuid5"
              },
              {
                "SegmentId": "uuid6"
              }
            ],
            "SummarizedSegment": "Weight loss."
          }
        ]
      }
    ]
  }
}
```

```
    ]
  },
  {
    "SectionName": "HISTORY_OF_PRESENT_ILLNESS",
    "Summary": [
      {
        "EvidenceLinks": [
          {
            "SegmentId": "uuid7"
          },
          {
            "SegmentId": "uuid8"
          },
          {
            "SegmentId": "uuid9"
          },
          {
            "SegmentId": "uuid10"
          }
        ],
        "SummarizedSegment": "The patient is seen today for a follow-up of weight
loss."
      },
      {
        "EvidenceLinks": [
          {
            "SegmentId": "uuid11"
          },
          {
            "SegmentId": "uuid12"
          },
          {
            "SegmentId": "uuid13"
          }
        ],
        "SummarizedSegment": "They report feeling more energy and craving
carbohydrates, sugar, and fat."
      },
      {
        "EvidenceLinks": [
          {
            "SegmentId": "uuid14"
          },
          {

```

```

        "SegmentId": "uuid15"
      },
      {
        "SegmentId": "uuid16"
      }
    ],
    "SummarizedSegment": "The patient is up to date on their mammogram and pap
smear."
  },
  {
    "EvidenceLinks": [
      {
        "SegmentId": "uuid17"
      },
      {
        "SegmentId": "uuid18"
      },
      {
        "SegmentId": "uuid19"
      },
      {
        "SegmentId": "uuid20"
      }
    ],
    "SummarizedSegment": "The patient is taking phentermine and would like to
continue."
  }
]
},
{
  "SectionName": "REVIEW_OF_SYSTEMS",
  "Summary": [
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid21"
        },
        {
          "SegmentId": "uuid22"
        }
      ],
      "SummarizedSegment": "Patient reports intermittent headaches, occasional
chest pains but denies any recent fevers or chills."
    }
  ],

```

```
{
  "EvidenceLinks": [
    {
      "SegmentId": "uuid23"
    },
    {
      "SegmentId": "uuid24"
    }
  ],
  "SummarizedSegment": "No recent changes in vision, hearing, or any
respiratory complaints."
}
],
{
  "SectionName": "PAST_MEDICAL_HISTORY",
  "Summary": [
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid25"
        },
        {
          "SegmentId": "uuid26"
        }
      ],
      "SummarizedSegment": "Patient has a history of hypertension and was
diagnosed with Type II diabetes 5 years ago."
    },
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid27"
        },
        {
          "SegmentId": "uuid28"
        }
      ],
      "SummarizedSegment": "Underwent an appendectomy in the early '90s and had a
fracture in the left arm during childhood."
    }
  ]
},
{
```

```
"SectionName": "ASSESSMENT",
"Summary": [
  {
    "EvidenceLinks": [
      {
        "SegmentId": "uuid29"
      },
      {
        "SegmentId": "uuid30"
      }
    ],
    "SummarizedSegment": "Weight loss"
  }
],
},
{
  "SectionName": "PLAN",
  "Summary": [
    {
      "EvidenceLinks": [
        {
          "SegmentId": "uuid31"
        },
        {
          "SegmentId": "uuid32"
        },
        {
          "SegmentId": "uuid33"
        },
        {
          "SegmentId": "uuid34"
        }
      ],
      "SummarizedSegment": "For the condition of Weight loss: The patient was
given a 30-day supply of phentermine and was advised to follow up in 30 days."
    }
  ]
}
]
```

Criptografia de dados em repouso para o AWS HealthScribe

O AWS HealthScribe fornece criptografia por padrão para proteger dados em repouso sigilosos do cliente usando chaves gerenciadas pelo Amazon S3.

- Chaves gerenciadas PELO Amazon S3 (SSE-S3): o AWS HealthScribe usa chaves gerenciadas pelo Amazon S3 por padrão para criptografar automaticamente arquivos intermediários. Não é possível visualizar, gerenciar nem usar chaves gerenciadas pelo Amazon S3, nem auditar o uso delas. No entanto, não é necessário realizar nenhuma ação nem alterar nenhum programa para proteger as chaves que criptografam os dados. Consulte mais informações em [SSE-S3](#).

A criptografia de dados em repouso por padrão ajuda a reduzir a sobrecarga operacional e a complexidade envolvidas na proteção de dados confidenciais. Ao mesmo tempo, ela permite que você crie aplicações seguras que atendam aos rigorosos requisitos regulatórios e de conformidade de criptografia.

Embora você não possa desabilitar essa camada de criptografia ou selecionar um tipo de criptografia alternativo, é possível adicionar uma segunda camada de criptografia sobre as chaves gerenciadas pelo Amazon S3 existentes escolhendo uma chave gerenciada pelo cliente ao criar um trabalho com o AWS HealthScribe.

- Chaves gerenciadas pelo cliente: o AWS HealthScribe é compatível com o uso de uma chave simétrica gerenciada pelo cliente que você cria, possui e gerencia para adicionar uma segunda camada de criptografia sobre a criptografia já existente de propriedade da AWS. Como você tem controle total dessa camada de criptografia, você pode realizar tarefas como:
 - Estabelecer e manter as políticas de chave
 - Estabelecer e manter concessões e políticas do IAM
 - Habilitar e desabilitar políticas de chaves
 - Alternar os materiais de criptografia de chaves
 - Adicionar etiquetas
 - Criar aliases de chaves
 - Programar a exclusão de chaves

Consulte mais informações em [customer managed key](#) no Guia do desenvolvedor do AWS Key Management Service.

Note

O AWS HealthScribe habilita automaticamente a criptografia em repouso usando chaves da AWS para proteger dados de identificação pessoal sem nenhum custo. No entanto, AWS KMS há cobrança aplicável pelo uso de uma chave gerenciada pelo cliente. Para obter mais informações sobre a definição de preço, consulte [Definição de preço do AWS Key Management Service](#).

Para obter mais informações sobre o AWS KMS, consulte [O que é o AWS Key Management Service](#).

Criação de uma chave gerenciada pelo cliente

Você pode criar uma chave simétrica gerenciada pelo cliente usando o AWS Management Console ou as APIs do AWS KMS. Para criar uma chave simétrica gerenciada pelo cliente, siga as etapas em [Creating symmetric customer managed key](#) no Guia do desenvolvedor do AWS Key Management Service.

As políticas de chaves controlam o acesso à chave gerenciada pelo seu cliente. Cada chave gerenciada pelo cliente deve ter exatamente uma política de chaves, que contém declarações que determinam quem pode usar a chave e como pode usá-la. Ao criar a chave gerenciada pelo cliente, você pode especificar uma política de chaves. Consulte mais informações em [Managing access to customer managed keys](#) no Guia do desenvolvedor do AWS Key Management Service.

Se estiver usando uma chave na mesma conta do perfil do IAM especificado como [DataAccessRoleArn](#) na solicitação [StartMedicalScribeJob](#), você não precisará atualizar a política de chave. Para usar a chave gerenciada pelo cliente em uma conta diferente como o `DataAccessRole`, você deve confiar no `DataAccessRoleArn` na política de chave para as seguintes ações:

- [kms:Encrypt](#): permite criptografar usando a chave gerenciada pelo cliente
- [kms:Decrypt](#): permite descriptografar usando a chave gerenciada pelo cliente
- [kms:DescribeKey](#): fornece os detalhes da chave gerenciada pelo cliente para permitir que o AWS HealthScribe valide a chave.

Veja a seguir um exemplo de instrução de política que você pode adicionar para conceder ao perfil do IAM permissões entre contas para usar a chave gerenciada pelo cliente:

```
"Statement" : [
```

```

{
  "Sid": "Allow access to the DataAccessRole for StartMedicalScribeJob",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/DataAccessRole"
  },
  "Action": [
    "kms:DescribeKey",
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource" : "*"
}
]

```

Independentemente de sua chave gerenciada pelo cliente e do `DataAccessRole` estarem na mesma conta ou em contas diferentes, o `DataAccessRole` precisará de permissões para realizar as ações mencionadas acima usando a chave gerenciada pelo cliente. Aqui está um exemplo de instrução de política que você pode adicionar ao `DataAccessRole`:

```

"Statement" : [
  {
    "Sid": "Allow role to perform AWS KMS actions for customer managed key",
    "Effect": "Allow",
    "Action": [
      "kms:DescribeKey",
      "kms:Encrypt",
      "kms:Decrypt"
    ],
    "Resource": "*"
  }
]

```

Consulte mais informações sobre [especificação de permissões em uma política](#) no Guia do desenvolvedor do AWS Key Management Service. Consulte mais informações sobre [solução de problemas de acesso à chave](#) no Guia do desenvolvedor do AWS Key Management Service.

Especificar uma chave gerenciada pelo cliente para o AWS HealthScribe

É possível especificar uma chave gerenciada pelo cliente como uma criptografia de segunda camada para solicitações `StartMedicalScribeJob`. Ao criar uma solicitação [StartMedicalScribeJob](#), você pode

especificar uma chave gerenciada pelo cliente incluindo o campo [OutputEncryptionKMSKeyId](#) na solicitação.

Contexto de criptografia do AWS KMS

O contexto de criptografia do AWS KMS é um mapa de texto simples, pares de chave-valor não secretos. Esse mapa representa dados autenticados adicionais, conhecidos como pares de contexto de criptografia, que fornecem uma camada adicional de segurança para seus dados. AWS O HealthScribe exige uma chave de criptografia simétrica para criptografar a saída do AWS HealthScribe em um bucket do Amazon S3 especificado pelo cliente. Para saber mais, consulte [Asymmetric keys in AWS KMS](#).

Ao criar pares de contexto de criptografia, não inclua informações confidenciais. O contexto de criptografia não é secreto. Ele é visível em texto simples nos logs do CloudTrail (para que você possa usá-lo a fim de identificar e categorizar as operações criptográficas). Seu par de contexto de criptografia pode incluir caracteres especiais, como sublinhados (), traços (-), barras (/ , \) e dois pontos (:).

Tip

Pode ser útil relacionar os valores do par de contexto de criptografia aos dados que estão sendo criptografados. Embora não seja obrigatório, recomendamos que você use metadados não confidenciais relacionados ao conteúdo criptografado, como nomes de arquivos, valores de cabeçalho ou campos de banco de dados não criptografados.

Para usar a criptografia de saída com a API, defina o parâmetro [KMSEncryptionContext](#) na operação [StartMedicalScribeJob](#). Para fornecer contexto de criptografia para a operação de criptografia de saída, o parâmetro [OutputEncryptionKMSKeyId](#) deve fazer referência a um ID de chave do AWS KMS simétrica.

É possível usar [chaves de condição do AWS KMS](#) com políticas do IAM para controlar o acesso a uma chave do AWS KMS de criptografia simétrica com base no contexto de criptografia que foi usado na solicitação de uma [operação criptográfica](#). Para ver um exemplo de política de contexto de criptografia, consulte [AWS KMS encryption context policy](#).

O uso do contexto de criptografia é opcional, mas recomendado. Consulte mais informações em [Encryption context](#).

Histórico do documento para Amazon Transcribe

- Última atualização da documentação: 13 de novembro de 2023

A tabela a seguir descreve mudanças importantes em cada versão do Amazon Transcribe. Para receber notificações sobre atualizações dessa documentação, você poderá se inscrever em um feed RSS.

Alteração	Descrição	Data
Atualização de recursos	Atualize o número máximo de alto-falantes em diarização para 30 em vez de 10.	10 de maio de 2024
Atualização de seção	Atualiza o resumo generativo de chamadas e adiciona detalhes da saída de erro.	30 de abril de 2024
Atualização de seção	Atualizações nas colunas de vocabulário personalizadas - IPA e. SoundsLike	30 de abril de 2024
Atualização de recursos	O Amazon Transcribe Call Analytics agora oferece suporte ao resumo generativo de chamadas.	29 de novembro de 2023
Atualização da seção	Atualize o novo formato de saída de edição de PII e identificação de idioma.	13 de novembro de 2023
Atualização de recurso	A diarização agora pode ser combinada com a identificação de canal.	6 de março de 2023

Atualização de recursos	A identificação de canal agora pode ser combinada com a diarização.	6 de março de 2023
Atualização da seção	IAM as melhores práticas foram atualizadas.	13 de fevereiro de 2023
Novos idiomas	Amazon Transcribe agora suporta vietnamita e sueco.	6 de dezembro de 2022
Novo atributo	Amazon Transcribe agora oferece suporte à análise de chamadas em tempo real.	28 de novembro de 2022
Atualização de recursos	A edição e a identificação de streaming agora estão disponíveis em hindi e tailandês.	11 de novembro de 2022
Atualização de seção	Novas categorias de PII estão disponíveis para edição e identificação de streaming.	14 de setembro de 2022
Atualização de seção	A seção de modelo de idioma personalizado foi revisada.	18 de junho de 2022
Atualização de seção	A identificação de idiomas em lote agora pode identificar vários idiomas por arquivo de áudio.	31 de maio de 2022
Atualização do guia	A Referência Amazon Transcribe da API agora é um guia independente.	1.º de abril de 2022

Novo capítulo	Uma nova tabela de comparação para Amazon Transcribe análises Amazon Transcribe médicas e de Amazon Transcribe chamadas está incluída.	21 de março de 2022
Novo capítulo	Um novo capítulo de exemplos de código do SDK foi incluído.	21 de março de 2022
Atualização de recurso	O Call Analytics agora fornece resumo de chamadas.	21 de março de 2022
Atualização de capítulo	O capítulo introdutório agora mostra casos de Amazon Transcribe uso.	21 de março de 2022
Atualização de capítulo	O capítulo “Conceitos básicos” foi atualizado para ser específico do método.	21 de março de 2022
Atualização de capítulo	O capítulo “Streaming” foi atualizado e reestruturado.	21 de março de 2022
Atualização de recurso	A identificação de idioma agora pode ser usada com vocabulários personalizados e filtros de vocabulário personalizados com transcrições de streaming.	11 de março de 2022
Novo evento	Há um novo tipo de evento: eventos de vocabulário.	7 de fevereiro de 2022
Atualização de seção	Foram feitas atualizações na seção de vocabulários personalizados.	20 de janeiro de 2022

Novo atributo	A identificação de idioma agora pode ser usada com transcrições de streaming.	23 de novembro de 2021
Novo atributo	Agora, a identificação de idioma pode ser usada com modelos de idioma personalizados, vocabulários personalizados, filtragem de vocabulário e edição de conteúdo.	29 de outubro de 2021
Novo atributo	Amazon Transcribe agora oferece suporte a modelos de linguagem personalizados com transcrições de streaming .	20 de outubro de 2021
Novo atributo	Amazon Transcribe agora pode gerar legendas para seus arquivos de vídeo.	16 de setembro de 2021
Novo atributo	Amazon Transcribe agora oferece suporte à redação e identificação de PII para streaming.	14 de setembro de 2021
Novo atributo	Amazon Transcribe agora oferece suporte ao contexto de AWS KMS criptografia para um nível adicional de segurança para seus Conta da AWS recursos.	10 de setembro de 2021

Novos idiomas	Amazon Transcribe agora suporta africâner, dinamarquês, chinês mandarim (tradicional), tailandês, inglês neozelandês e inglês sul-africano.	26 de agosto de 2021
Novo atributo	Amazon Transcribe agora oferece suporte à marcação de recursos.	24 de agosto de 2021
Novo atributo	Amazon Transcribe agora oferece suporte ao Call Analytics para trabalhos de transcrição em lote.	4 de agosto de 2021
Novo atributo	Amazon Transcribe agora suporta o uso de vocabulários personalizados com modelos de linguagem personalizados em lote.	12 de maio de 2021
Novo atributo	Amazon Transcribe agora suporta estabilização parcial de resultados para transcrição de streaming.	11 de maio de 2021
Novo atributo	Amazon Transcribe agora oferece suporte ao inglês australiano, inglês britânico, hindi e espanhol dos EUA para modelos de idiomas personalizados.	19 de março de 2021
Novo atributo	Amazon Transcribe agora suporta codecs OGG/OPUS e FLAC para streaming de transcrição de áudio.	24 de novembro de 2020

Novos idiomas	Amazon Transcribe adiciona suporte para italiano e alemão para streaming de transcrição de áudio.	4 de novembro de 2020
Região da AWS expansão	Amazon Transcribe agora está disponível em Frankfurt (eu-central-1) e Londres (eu-west-2).	4 de novembro de 2020
Novo atributo	Amazon Transcribe adiciona suporte para endpoints VPC de interface na transcrição em lote.	9 de outubro de 2020
Novo atributo	Amazon Transcribe adiciona suporte para identificação de canais em streaming.	17 de setembro de 2020
Novo atributo	Amazon Transcribe adiciona suporte para identificação automática de idioma na transcrição em lote.	15 de setembro de 2020
Novo atributo	Amazon Transcribe adiciona suporte para particionamento de alto-falantes em streaming.	19 de agosto de 2020
Novo atributo	Amazon Transcribe adiciona suporte para modelos de linguagem personalizados.	5 de agosto de 2020
Novo atributo	Amazon Transcribe adiciona suporte para endpoints VPC de interface em streaming.	26 de junho de 2020

Novo atributo	Amazon Transcribe adiciona suporte para filtragem de vocabulário em streaming.	20 de maio de 2020
Novo atributo	Amazon Transcribe adiciona suporte para a edição automática de informações de identificação pessoal.	26 de fevereiro de 2020
Novo atributo	Amazon Transcribe adiciona suporte para criar um vocabulário personalizado de palavras para filtrar a partir de uma transcrição.	20 de dezembro de 2019
Novo atributo	Amazon Transcribe adiciona suporte para trabalhos de transcrição em fila.	19 de dezembro de 2019
Novos idiomas	Amazon Transcribe adiciona suporte para árabe do Golfo, hebraico, japonês, malaio, alemão suíço, telugu e turco.	21 de novembro de 2019
Região da AWS expansão	Amazon Transcribe agora está disponível na Ásia-Pacífico (Tóquio) (ap-northeast-1).	21 de novembro de 2019
Novo atributo	Amazon Transcribe adiciona suporte para transcrições alternativas.	20 de novembro de 2019
Novos idiomas	Amazon Transcribe adiciona suporte para holandês, persa, indonésio, inglês irlandês, português, inglês escocês, tâmil e inglês galês.	12 de novembro de 2019

Novo idioma	Amazon Transcribe agora suporta transcrição de streaming para inglês australia no (en-AU).	25 de outubro de 2019
Região da AWS expansão	Amazon Transcribe agora está disponível na China (Pequim) (cn-north-1) e na China (Ningxia) (cn-north west-1).	9 de outubro de 2019
Novo atributo	Amazon Transcribe permite que você forneça seus próprios KMS key para criptografar seus arquivos de saída de transcrição. Para obter mais informações, consulte o KeyId parâmetro OutputEncryptionKMS da StartStreamTranscription API.	24 de setembro de 2019
Novos idiomas	Amazon Transcribe adiciona suporte para chinês (mandarim), simplificado, China continental e russo.	23 de agosto de 2019
Novo atributo	Amazon Transcribe adiciona suporte para streaming de transcrição de áudio usando o WebSocket protocolo.	19 de julho de 2019
Novo atributo	AWS CloudTrail agora registra eventos para a StartStreamTranscription API.	19 de julho de 2019

Região da AWS expansão	Amazon Transcribe agora está disponível no Oeste dos EUA (Norte da Califórnia) (us-west-1).	27 de junho de 2019
Novo idioma	Amazon Transcribe adiciona suporte para o árabe padrão moderno.	28 de maio de 2019
Novo atributo	Amazon Transcribe agora transcreve palavras numéricas em números para o inglês americano. Por exemplo, “quarenta e dois” é transcrito como “42”.	23 de maio de 2019
Novo idioma	Amazon Transcribe adiciona suporte para hindi e inglês indiano.	15 de maio de 2019
Novo SDK	O AWS SDK para Amazon Transcribe C++ agora oferece suporte.	8 de maio de 2019
Novo idioma	Amazon Transcribe adiciona suporte para espanhol.	19 de abril de 2019
Região da AWS expansão	Amazon Transcribe agora está disponível na UE (Frankfurt) (eu-central-1) e na Ásia-Pacífico (Seul) (ap-northeast-2).	18 de abril de 2019
Novo idioma	Amazon Transcribe adiciona suporte para transcrição de streaming em inglês britânico, francês e francês canadense.	5 de abril de 2019

Novo atributo	O AWS SDK for Ruby V3 agora suporta Amazon Transcribe	25 de março de 2019
Novo atributo	Amazon Transcribe permite vocabulários personalizados, que são listas de palavras específicas que você Amazon Transcribe deseja reconhecer na entrada de áudio.	25 de março de 2019
Novos idiomas	Amazon Transcribe adiciona suporte para alemão e coreano.	22 de março de 2019
Novo idioma	Amazon Transcribe agora suporta transcrição de streaming para espanhol dos EUA (es-US).	7 de fevereiro de 2019
Região da AWS expansão	Amazon Transcribe agora está disponível na América do Sul (São Paulo) (sa-east-1).	7 de fevereiro de 2019
Região da AWS expansão	Amazon Transcribe agora está disponível na Ásia-Pacífico (Mumbai) (ap-south-1), Ásia-Pacífico (Cingapura) (ap-south east-1), UE (Londres) (eu-west-2) e UE (Paris) (eu-west3).	24 de janeiro de 2019
Novos idiomas	Amazon Transcribe adiciona suporte para francês, italiano e português do Brasil.	20 de dezembro de 2018

Novo atributo	Amazon Transcribe agora suporta a transcrição de fluxos de áudio.	19 de novembro de 2018
Novos idiomas	Amazon Transcribe adiciona suporte para inglês australiano, inglês britânico e francês canadense.	15 de novembro de 2018
Região da AWS expansão	Amazon Transcribe agora está disponível no Canadá (Central) (ca-central-1) e na Ásia-Pacífico (Sydney) (ap-southeast-2).	17 de julho de 2018
Novo atributo	Agora é possível especificar seu próprio local para armazenar a saída de um trabalho de transcrição.	11 de julho de 2018
Novo atributo	Adicionado AWS CloudTrail e Amazon CloudWatch Events integração.	28 de junho de 2018
Novo atributo	Amazon Transcribe adiciona suporte para vocabulários personalizados.	4 de abril de 2018
Novo guia	Esta é a primeira versão do Guia do desenvolvedor do Amazon Transcribe .	29 de novembro de 2017

AWS Glossário

Para obter a terminologia mais recente da AWS, consulte o [glossário da AWS](#) na Referência do Glossário da AWS.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.