



AWS Academy Natural Language Processing
Module 07 Student Guide
Version 0.1.0
200-ACMNLP-01-EN-SG

© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved.

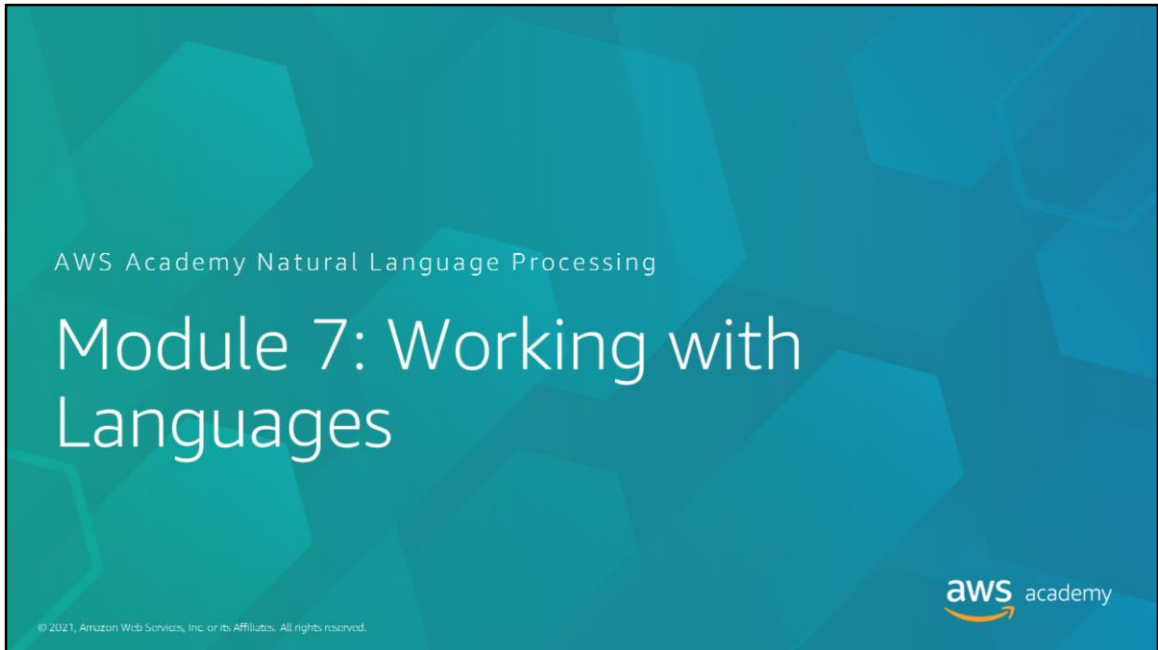
This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited.

All trademarks are the property of their owners.

Contents

[Module 7: Working with Languages](#)

4



Welcome to Working with Languages.

Module overview



Sections

1. Working with language issues
2. Detecting and translating languages
3. Transcribing and vocalizing text with AWS services

Labs

- Challenge Lab: Implementing a Multilingual Solution

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

2

In this module, you will learn about the following:

- 1. Working with language issues** – In this section, you will learn about some of the common issues that you might have when working with languages.
- 2. Detecting and translating languages** – In this section, you will get an overview of tools that you can use to detect and translate language.
- 3. Working with languages** – In this section, you will get an overview of Amazon Transcribe and Amazon Polly, and you learn about some of their common use cases.

Challenge Lab: Implementing a Multilingual Solution – In this lab, you will use Amazon Transcribe to extract text from an audio file. You will then use Amazon Transcribe to translate the text into another language and use Amazon Polly to generate a new audio file in the new language.

Module objectives



At the end of this module, you should be able to:

- Describe the challenges of working with languages
- Identify the predominant language of a text with Amazon Comprehend
- Describe common use cases for Amazon Translate, Amazon Polly, and Amazon Transcribe
- Implement a solution for transcribing and translating text

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

5

At the end of this module, you should be able to:

- Describe the challenges of working with languages
- Identify the predominant language of a text with Amazon Comprehend
- Describe common use cases for Amazon Translate, Amazon Polly, and Amazon Transcribe
- Implement a solution for transcribing and translating text

Module 7: Working with Languages

Section 1: Working with language issues

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



In this section you will learn about several common issues you will see in NLP applications.

Challenges: Understanding context



- Context objectives
 - Who is speaking?
 - What are they speaking about?
 - How do they feel about the subject?
 - Why do they feel that way?
- Context analysis
 - N-grams
 - Noun phrase extraction
 - Theme extraction



© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

5

As you have seen throughout this course, working with languages is a difficult problem for machine learning. One of the most challenging aspects is that the meaning of words can change dramatically, depending on context. To correctly extract context, your natural language processing (NLP) application must understand the following key factors:

- Who is speaking? For example, is this a voice of authority? Is it an inquiry from a customer?
- What is the subject of the text? For example, is the text about a customer issue or a medical diagnosis?
- How does the speaker or writer feel about what they are saying? Sentiment analysis comes into play here.
- Why do they feel that way? This aspect is one of the most difficult parts of extracting context. For example, if you detect that a movie review is negative, can you examine the text to discover why the viewer gave a negative review?

Context analysis applies techniques to help answer some of these questions. You start your context analysis by determining the subject of the text. N-grams are a good way to identify the subject because you can use them to find the most common groups of words in the text.

Noun phrase extraction is another technique for determining the subject of a text. Noun phrases are a combination of nouns with another part of speech. Examples of such combinations include a noun plus an adjective, such as “hairy dog,” or a noun plus a verb, such as “spinning top.”

Nouns and noun phrases are useful for understanding the subject and sentiment of text. In

addition, to fully grasp the context, you must identify themes within a text. You can identify themes by using theme extraction. Theme extraction combines N-grams or noun phrases with a relevancy score to determine the most important themes in the text. One scoring system that you can use is called *lexical chaining*. Lexical chaining ties noun phrases across multiple sentences to extract the key themes from a text.

Challenges: Language structure



- Syntax: The way that words are structured in a sentence
 - Word order and directionality
 - Subject, object, verb relationship
 - Possession and plurals
 - Gender agreement for some languages
- Semantic and syntax analysis
 - Semantic parsing
 - Statistical models
- Low-resource languages

English	The dog has black fur.
Spanish	El perro tiene pelaje negro.
English	Look at the dog's long tail.
German	Schau dir den langen Schwanz des Hundes an.
English	Look at the dog.
English	Look at the dogs.
Albanian	Shiko të gjithë qenin.
Albanian	Shiko të gjithë qentë.

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

b

As with any machine learning problem, choosing the right algorithm is a function of your business goal. To choose between Latent Dirichlet Allocation (LDA) and Neural Topic Model (NTM), you can consider two metrics: perplexity and coherence.

One of the biggest challenges that you face with any NLP application is how to deal with various sentence structures. For example, syntax is not consistent for various languages. Some of the most obvious variations include the following:

- **Word order:** In English, adjectives generally precede the nouns that they describe. This word order is not the case for many other languages; for example, Spanish.
- **Expression of possession or plurals:** Possession in English is often indicated with an apostrophe followed by an *s*. For many other languages, possession is indicated by including additional words. Expression of plurals is another factor that changes between languages. Whereas many English words create a plural by adding an *s*, other languages change multiple letters. In the Albanian example that you see here, the *in* changes to *të*.

Semantic analysis is the process of analyzing language to extract meaning from text. To extract meaning, semantic analysis combines the analysis based on language syntax with a layer that compares semantics across languages. These systems are known as semantic parsers.

Developing a statistical model for understanding semantics requires a large dataset of

validated translations in the source language. Assembling a dataset for less commonly spoken languages (known as *low-resource languages*) is a significant challenge for all NLP applications.

Challenges: Vocabulary and word issues



- Specialized vocabulary
 - Out of vocabulary (OOV) problem: Dealing with words that are not in your working vocabulary
- Homonyms
 - Homograph: *wind* and *wind*
 - Homophone: *hear* vs. *here*
- False cognates
 - English: *sensible* -> *reasonable*
 - Spanish: *sensible* -> *sensitive*
- Complex morphologies
 - Turkish: Base words change meaning by adding suffixes and prefixes



© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Vocabulary is another challenge area for working with language. One estimate puts the number of English words in current usage at over 170,000. Other languages have similar word counts, or even larger, depending on the unit of measure. In Module 4, you saw that one potential problem for your NLP application is words that are not recognized in your working vocabulary. This is known as an *out of vocabulary* (OOV) problem.

Other challenges include homonyms, which have two different types. First, you can encounter homographs, which have two words that are spelled the same way but have different pronunciations and meanings. For example, the sentence “I had to wind my way through the wind” contains two uses (and two different pronunciations) of the word *wind*. You can also encounter homophones, which have two words that sound the same but are spelled differently and also have different meanings. For example, the sentence “I can only hear you when you are here” contains the homophones *hear* and *here*.

Another problem you might run into is *false cognates*. False cognates are words that sound the same in two languages but have different meanings. For example, *sensible* in English means reasonable, whereas the word *sensible* in Spanish means sensitive. Languages that are derived from Latin contain many examples of false cognates.

Finally, some languages, such as Turkish and Finnish, have complex morphologies.

Morphology in language describes how words are formed by changing either prefixes or suffixes. Both Turkish and Finnish have a relatively small number of base words, which are then turned into a larger vocabulary by adding prefixes and suffixes.

Challenges: Characters and spelling



- Various character sets
 - Chinese-Japanese-Korean (CJK) characters
- Nonstandard spelling
 - Arabic, Hindi
 - Slang, shorthand in texts
- Character embedding
 - Similar to word embedding
 - Groups of letters that are assigned to a vector
 - Similar character vectors are grouped together

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

8



Most NLP systems are based on evaluating words or phrases. However, the best solution to some of the issues that you just learned about is to evaluate language at the individual character level. Yet, this approach has its own set of problems.

The vast array of character sets is a significant problem. For example, the International Standards Organization (ISO) has defined 15 standard encodings for various alphabets. Chinese-Japanese-Korean (CJK) characters further complicate character processing due to variations in how characters are formed.


Nonstandard spellings are another reason to employ character evaluation. For example, many countries where Arabic is the primary language have two different variations of the language, one standard and one nonstandard. This situation can make evaluating at the word or sentence level quite difficult.

Character embedding is similar to word embedding, which you learned about in Module 3. With character embedding, each letter in a word is assigned a vector. The model that is produced from these vectors is then trained to group similar vectors.


Language use cases




Machine translation



Automatic transcription



Subtitle generation



Text analysis

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

The following list presents some common use cases for applying machine learning to language:

- **Machine translation:** Using machines to translate between human languages has been a major problem in computer science for many years. In fact, machine translation was one of the earliest goals of computing. Recent advances have made it possible to have real-time translation in some of the most commonly spoken languages.
- **Subtitle generation:** Creating subtitles for movies and videos has been greatly streamlined through the advent of machine learning-based transcription services.
- **Automatic transcription:** Converting spoken language to a written form has many applications, from medical records to legal or other business settings.
- **Text analysis:** The explosion of text-based interactions in recent years has created a major opportunity for businesses to learn more about their customers and their own operations. Capturing and analyzing text is being used for analyzing customer support requests, reviewing and categorizing medical diagnoses, and many other text-based records.

Section 1: Summary



- Language challenges
 - Understanding context
 - Language structure
 - Vocabularies
 - Character sets
 - Machine translations
- Language use cases
 - Transcription
 - Translation
 - Text and voice

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

10

In this section, you looked at some of the challenges of working with language. These challenges include the following:

- Understanding context
- Variance in language structure for different languages
- Problems with large vocabularies and differences in syntax

Finally, you looked at some of the more common use cases for transcription, translation, and text and voice applications.

Module 7: Working with Languages


Section 2: Detecting and translating languages

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



In this section, you look at tools for detecting and translating languages.

Language detection tools



- Amazon Comprehend
 - Uses a pretrained model
 - Can detect more than 100 different languages
- Python packages
 - langdetect
 - spaCy
 - fastText

Entities | Key phrases | **Language** | PII | Sentiment | Syntax

Analyzed text

Er was geen mogelijkheid om een wandeling te maken die dag. We hadden inderdaad een uur in de ochtend rondgelopen in de bladerloze struiken; maar sinds het diner (mevrouw Reed, toen er geen gezelschap was, vroeg gegeten) had de koude winterwind zo somber en een regen zo doordringend, dat er nu geen sprake was van verdere oefening in de buitenlucht..

▼ Results

Language

Dutch, nl
0.99 confidence

langdetect example

```
pip install langdetect
from langdetect import detect
detect("this is some text")

Result 'en'
```

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

12

You have already learned about several use cases for Amazon Comprehend in this course. You can also use the service to detect the language of a text. Amazon Comprehend uses a pretrained model and can detect more than 100 different languages. When Amazon Comprehend returns the detected language, it also returns a confidence score for the selection.

In addition to Amazon Comprehend, you can use several Python packages to detect the language of a text. The langdetect, spaCy, and fastText packages all include libraries that you can call to detect a language.

Translation tools



- Amazon Translate
 - Uses a pretrained model
 - Can detect more than 100 different languages
- Python packages
 - langdetect
 - translate
 - fastText

Translation

Source language

Auto (auto)

από τις πιο θορυβές αρχές της επέμεναν να την παραλάβουμε, για καλό ή για κακό, μόνο σε υπερθετικό βαθμό σύγκρισης.

600 characters, 1079 of 5000 bytes used.

[Info](#)

Detected language: Greek (el)

```
import boto3
translate = boto3.client(service_name='translate',
                        region_name='us-east-1', use_ssl=True)
result = translate.translate_text(Text="Hello, World",
                                SourceLanguageCode="en", TargetLanguageCode="es")
print('TranslatedText: ' + result.get('TranslatedText'))
```

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

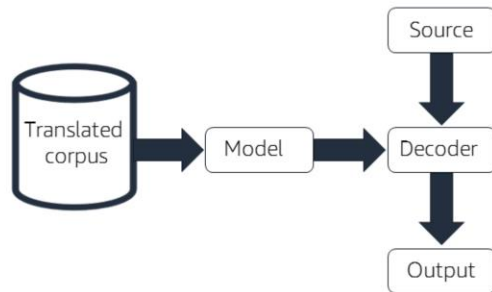
15

Amazon Translate is a neural machine translation (NMT) system. The system uses a pretrained model and can detect 100 different languages. The slide shows an example of language detection in the Amazon Translate console. You can also generate translations by using the AWS Command Line Interface (AWS CLI) or the application programming interface (API).

Statistical machine translation (SMT)



- SMT systems are based on applying statistical models to predict translations
- Words and phrases are the units of translation
- Challenges
 - Understanding context
 - Works better for some language pairs
 - Languages with dramatic differences in word order



© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

14

Machine translation has two approaches: statistical machine translation (SMT) and neural machine translation (NMT).

Statistical machine translation (SMT) starts by creating a model of a source language. The model is based on a large set of previously translated documents. After you have the model, you can apply it to new texts in the source language to predict the translation in the target language. These predictions are based on probability theory.

SMT is based on translations of individual words and phrases. The SMT system builds a translation table, which lists the frequency of word combinations in the source language collection. Phrase translations with a higher frequency in the source collection have a greater probability of being correctly translated than translations with a lower frequency.

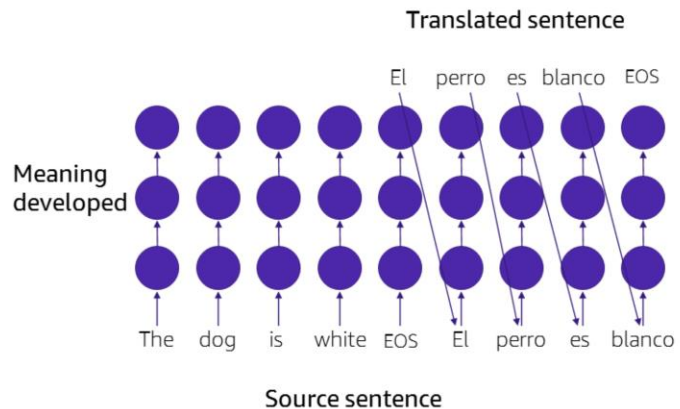
SMT systems have several problems:

- They are not well suited for understanding the influence of context on meaning.
- Accuracy depends on language pairs. For example, English and Spanish might work better than English and Mandarin.
- SMT systems do not work well with languages that have dramatic differences in word order and sentence length.

Neural machine translation (NMT)



- NMT systems use neural networks to learn a model for translation
 - Built with an encoder-decoder model
 - Most use recurrent neural networks
- Challenges
 - Long sequences of text
 - Large vocabularies
 - Long training times
 - Low-resource languages



© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

15

Neural machine translation (NMT) systems use neural networks to learn a model based on previous translations. As you learned in Module 2, neural networks include two sets of nodes: one for an encoder layer and one for a decoder layer. Most NMT systems are built with recurrent neural networks. NMT systems can capture the context in a vector and then apply that vector to multiple language pairs. Therefore, NMT systems are better at understanding context than SMT systems.

NMT systems are generally an improvement over SMT systems, which were developed before neural networks and were in wide-scale commercial use. However, they still present several challenges:

- Long sequences of text, for example very long sentences, can cause the network to *forget* the earlier part of the sequence. You can mitigate this problem by introducing an attention layer, which was discussed in Module 2.
- Large or specialized vocabularies are still a problem for NMT.
- NMT systems are resource intensive and require long training times.
- Languages that have no translated models are known as *low-resource languages*.

Section 2: Summary



- Language detection tools
 - Amazon Comprehend
 - Python packages
- Translation tools
 - Amazon Translate
 - Python packages
- Translation methods
 - Statistical machine translation
 - Neural machine translation

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

1b

In this section, you looked at tools for detecting and translating language. You also learned about two different methods for translating text: statistical machine translation and neural machine translation.

Module 7: Working with Languages

Section 3: Transcribing and vocalizing text with AWS services

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

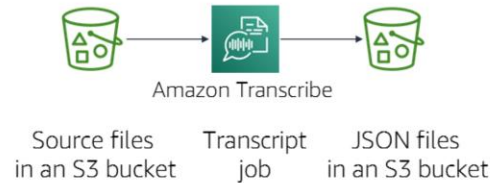


In this section, you look at two more Amazon machine learning services that you can use when working with language: Amazon Transcribe and Amazon Polly.

Transcribing text with Amazon Transcribe



- Convert audio files to text
 - Store files in Amazon Simple Storage Service (Amazon S3)
 - Invoke a job from the console, AWS CLI, or API
- Convert streaming data
 - Real-time translation
 - Invoke text processing from the transcription
- Identify speakers
- Transcribe from multiple channels



```

{"jobName":"Testtranscribe","accountId":"", "results":{"transcripts":[{"transcript":"test. Hello? Hello. This is a test test test test."}], "items":[{"start_time":"0.0","end_time":"0.47","alternatives":[{"confidence":"0.9875","content":"test"},"type":"pronunciation"}, {"alternatives":[{"confidence":"0.0","content":"."},"type":"punctuation"}, {"start_time":"0.48","end_time":"1.29","alternatives":[{"confidence":"1.0","content":"Hello"},"type":"pronunciation"}, {"alternatives":[{"confidence":"0.0","content":"?"},"type":"punctuation"}, {"start_time":"1.3","end_time":"1.79","alternatives":[{"confidence":"1.0","content":"Hello"},"type":"pronunciation"}, {"alternatives":[{"confidence":"0.0","content":"."},"type":"punctuation"}, {"start_time":"1.8","end_time":"2.46","alternatives":[{"confidence":"0.9154","content":"Hello"},"type":"pronunciation"}, {"alternatives":
  
```

Sample output

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

18



You can use Amazon Transcribe to convert spoken language into text. You store the audio files in Amazon Simple Storage Service (Amazon S3) and then invoke a transcribe job from the AWS Management Console. Alternatively, you can write a script to use the AWS CLI to store the output in Amazon S3

You can also create real-time transcripts by sending streaming data to Amazon Transcribe. Amazon Transcribe supports two different protocols for streaming data: WebSockets and HTTP/2.


Amazon Transcribe returns text as JavaScript Object Notation (JSON) files. These files contain metadata about the audio, such as start and end time, and speaker identification if more than one voice is present.

Amazon Transcribe can also create transcripts from multiple audio channels. If you have multiple channels, Amazon Transcribe creates separate transcriptions for each channel and then combines them into a single JSON file.


Amazon Transcribe use cases




Speech-to-text notifications



Call center analysis



Medical and legal transcriptions



Real-time translations

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

19

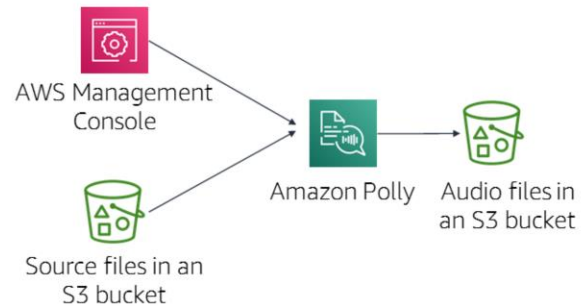
The following are example use cases for Amazon Transcribe:

- Generate text messages based on audio files.
- Create a written transcript from spoken records of medical or legal experts.
- Store call center records in Amazon S3 and then export the transcript output to be analyzed.
- Create a real-time translation engine by integrating the service with Amazon Comprehend and Amazon Translate.

Converting text to voice with Amazon Polly



- Input from text files
- Real-time processing or batch processing
- Custom voices
- Neural text-to-speech (TTS) for higher quality output
- Custom lexicons for specialized terminology
- Speech Synthesis Markup Language (SSML) support



© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

20

With Amazon Polly, you can convert text to audio for various applications. You can either copy and paste into the AWS Management Console or read text files from Amazon S3 for batch processing of longer texts. You can also customize voices by selecting from various speaking styles. For example, multiple English-speaking voices are available.

In addition to the standard voices, you can select one of the neural text-to-speech (TTS) voices. These voices are generated by using a sequence-to-sequence network and have a more natural sound.

You can add custom lexicons to support specialized vocabularies and control speech patterns by adding Speech Synthesis Markup Language (SSML) tags. For example, you can add an emphasis tag for words or phrases that you want the voice to stress. You can also pause speaking and control volume, speaking rate, and pitch.

Modifying Amazon Polly output



- Common SSML tags

Action	Tag
Add a pause	<break>
Emphasize words	<emphasis>
Control volume, rate, and pitch	<prosody>

```
<speak>Hi, I'm going to demonstrate some  
SSML tags.<break> That was a short pause. You  
can also create longer pauses, like this <break  
time="3s"/.  
With the prosody tag, you can set the volume  
<prosody value="loud">like  
this.</prosody></speak>
```

- Speech marks
 - Synchronize speech with video
- Custom lexicons

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



21

You can control Amazon Polly output by adding SSML tags to the text. The `<break>` tag adds a pause in the speech. You can specify the length of the pause by adding a number of seconds, or you can add preset values, such as *weak* or *strong*. Find the complete list of SSML tags in the Amazon Polly documentation.


If you are coordinating your speech output with visual output, you can have Amazon Polly insert speech marks. Speech marks are metadata that specify when the sounds in the audio output begin and end. For example, you might have Amazon Polly export speech marks for every sentence or word in an audio file. You could then use these speech marks to synchronize the output with animation or to dub a live action video.

The text files that you are working with might contain a large quantity of specialized vocabulary. In this case, you can create a custom lexicon that instructs Amazon Polly how to pronounce these specialized words.


Amazon Polly use cases




Audio on websites



Voice response for contact centers



Animations with speech



Vocalized alerting

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

22

The following list contains a few examples of how you can use Amazon Polly to add voices to your applications.

Make websites more accessible by adding voices.

Improve contact center interfaces with voice response.

Add speech to animations for digital learning assets.

Make alerting systems more accessible by adding a voice option to text notifications

Module 7 – Challenge Lab 1: Implementing a Multilingual Solution

25



© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

You will now complete Module 7 – Challenge Lab 1: Implementing a Multilingual Solution.

Section 3: Summary

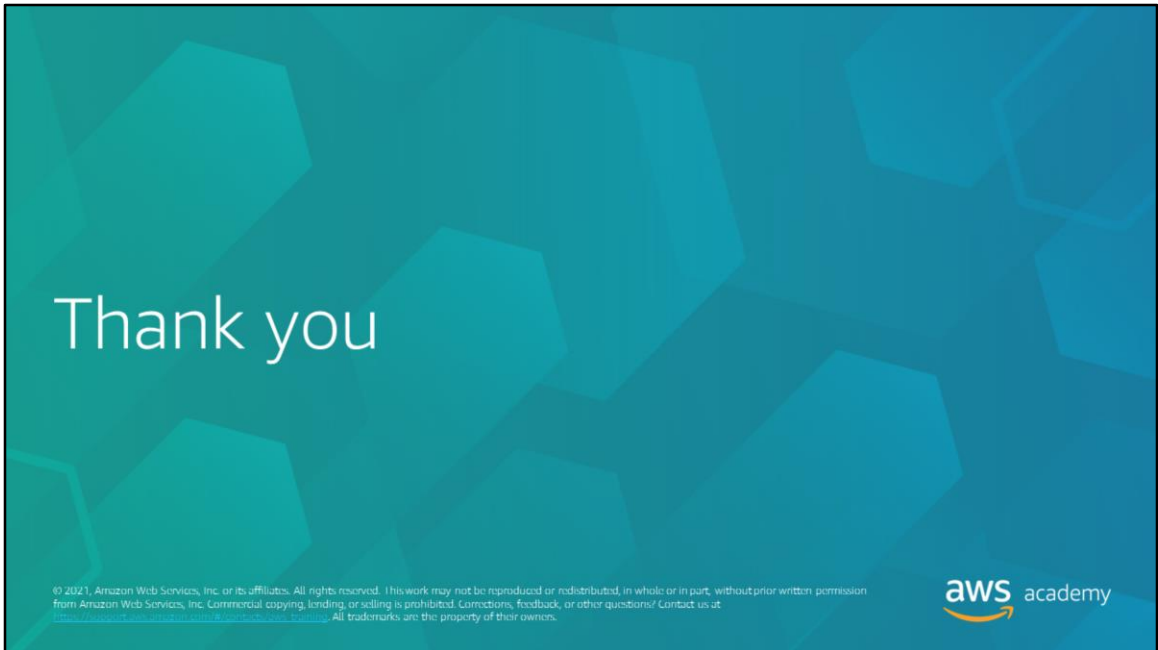


- Amazon Transcribe
 - Overview
 - Use cases
- Amazon Polly
 - Overview
 - Use cases

© 2021, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

24

In this section, you learned about the features and use cases for Amazon Transcribe and Amazon Polly.



Thank you for completing this module.