**CHAPTER 4**

**Describe features of Natural Language Processing (NLP) workloads on Azure**

*Natural Language Processing (NLP)* is the processing of text and speech to extract meaning. NLP can interpret written text and spoken audio and provide detail and understanding about the language used in a computer-readable form.

Natural Language Processing services are provided by Azure Cognitive Services. The concepts involved in NLP will be outlined with use cases, followed by how to use the various Azure Cognitive Services language services.

Language has been a major focus for AI. For many organizations, processing is centered around text and documents and there are large gains to be made by applying AI. There are large amounts of text held in computer-readable formats that can be used to train models. AI can assist users in handling a significant volume of text data and audio streams. For instance, AI can:

* Mine text for patterns and trends.
* Transform unstructured text into more structured formats.
* Generate medical codes by analyzing healthcare documents for specific phrases and terms.
* Sift social media, blog feeds, and webpages to monitor trends.
* Interpret requests and decide on the best action to take—for example, digital assistants.
* Translate text from one language into other languages.

This chapter describes the prebuilt language-related AI capabilities provided by Azure Cognitive Services and explains how to use these language services.

**Skills covered in this chapter:**

* [Skill 4.1: Identify features of common NLP workload scenarios](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04lev1sec1)
* [Skill 4.2: Identify Azure tools and services for NLP workloads](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04lev1sec2)

**Skill 4.1: Identify features of common NLP workload scenarios**

Natural Language Processing is concerned with understanding written and spoken language. There are countless scenarios for using NLP in applications for organizations across many industries.

A human being can instinctively read and understand the meaning of a piece of unstructured text. NLP does not understand the text in the same way as a human; NLP is the process of gaining insights into text and extracting some meaning from that text.

A major aspect of the Microsoft Azure AI Fundamentals certification is on the capabilities and features of Natural Language Processing and how language services can be applied in different scenarios. This requires you to understand the use cases for Natural Language Processing and to be able to differentiate the various language services in Microsoft Azure.

This skill covers how to:

* Describe Natural Language Processing
* Describe language modeling
* Describe key phrase extraction
* Describe named entity recognition
* Describe sentiment analysis
* Describe speech recognition and synthesis
* Describe translation

**Describe Natural Language Processing**

Natural Language Processing (NLP) is the process of applying AI algorithms and models to text in documents, email messages, and other sources, as well as to speech to extract attributes about the text and gain insights from the text.

NLP can be used to classify documents or summarize documents by identifying the subjects in the document. The output of NLP can be used for further text-based processing or searching.

NLP can extract key phrases from a document. NLP can recognize questions and intent from text and extract those into requests and actions. NLP can perform sentiment analysis on a document to decide how positively or negatively the language is used.

NLP can interpret spoken language and synthesize speech responses.

Image ***EXAM TIP***

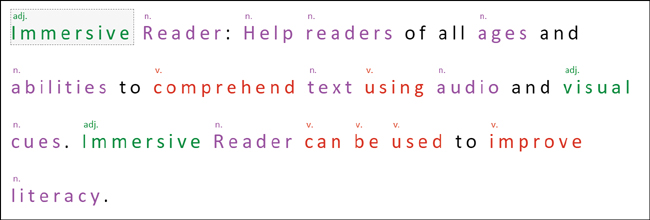
You need to be able to map the NLP workloads on the scenarios presented or identify which type of NLP workload applies to the requirements described.

**Text Analytics techniques**

NLP is concerned with applying the following processing techniques to extract meaning and gain insights:

* Analysis of text
  + **Tokenize**   Splitting text into words and phrases.
  + **Statistical**   Analyzing the terms used, including the frequency of the appearance of individual words.
  + **Frequency**   As well as the frequency of individual words, identifying the frequency of phrases.
  + **PosTag (Part of speech tagging)**   Assigning parts of speech (noun, verb, or ­adjective) to each word.
  + **Sentiment analysis**   Scoring the text for sentiment, as having a positive or negative feeling.
  + **Language detection**   Detecting the predominate language used in the text.
* Language modeling
  + **Semantic modeling**   Identifying the relationships between words.
  + **Named entity recognition (NER)**   Identifying objects (places, dates, or quantities) in the text.
  + **Topic detection**   Combining entities into topics to describe the important topics present in the text.
* Analysis of speech
  + **Conversion of audio into text**   Analyzing and interpreting speech and converting into text.
  + **Conversion of text into audio**   Analyzing text, identifying phrases, and synthesizing those phrases into spoken audio.
* Translation
  + Automatic translation between languages for both text and speech.

[Figure 4-1](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig01) shows a screenshot of the Immersive Reader tool. This demonstrates the use of speech tagging. Each word is identified with a tag (adjective, noun, or verb) and is color coded.



**FIGURE 4-1** Immersive Reader

**Use cases for NLP**

There are many use cases for NLP. Some examples are listed next.

A company can search for terms in documents related to their products. For instance, a patent attorney could use NLP to look for examples of prior use when evaluating new patents, or to protect their existing patents by searching for terms in their patents against patents submitted by others.

An organization can monitor social media posts for sentiment and then flag negative sentiment. An example would be an airline that monitors Twitter for posts that mention the airline’s name. Any negative tweet can be flagged, and the follower count of the poster used to determine the level of response required.

A call center can use NLP to analyze phone calls. NLP can transcribe the call and identify which voice is speaking.

Organizations can classify emails in customer support. Emails can be prioritized for the urgency of response, if the email is SPAM, or even if an email requires a response at all.

Now, we will examine the major NLP workloads.

**Describe language modeling**

A common use of NLP is having conversations with humans through chatbots and personal digital assistants. To enable computers to have human-like interactions such as these requires the computer to interpret the meaning of the text supplied, understand what is being requested, and what the request is about.

***NOTE*   LANGUAGE MODEL**

A language model is a core component in NLP. The language model uses the statistical techniques described previously to analyze the structures and patterns in text. For instance, a language model will look at the probability of a sequence of words. This is more than extracting the nouns and verbs in a sentence. It is having a model of the language to be able to understand the text supplied. It is more than rules for grammar; humans do not always follow the complex rules for grammar, and these rules vary over time and context.

Language models have been trained to learn the structure of a language by analyzing enormous amounts of text. Language modeling aims to interpret the intent from a text statement and extract key information to discover the overall meaning from the text.

Language modeling is at the heart of creating NLP-based solutions and forms the basis for the other workloads described in this section. Converting a command into smart actions is an example of language modeling. Language modeling interprets the purpose of a text command and turns that command into an intent that can be converted into a smart action for a device to run.

Language modeling does not detect the language in which the text is written. Detecting the language that the text is written in is language detection, not language modeling. Language modeling can be performed in many languages but only one language at a time. Similarly, language modeling does not determine the emotion in a text statement. Determining the emotion in a text statement is sentiment analysis, not language modeling.

**Describe key phrase extraction**

*Key phrase extraction* is the evaluation of a piece of text to identify the main talking points, or popular mentions, contained in the text. Key phrase extraction returns a list of talking points. Each talking point is normally a single word or a short phrase.

An example use for key phrase extraction is for website reviews. You can extract key phrases from all the reviews and then add these elements as tags to allow other website users to filter reviews. For example, for the review text, "The food was delicious and there were wonderful staff," the key phrase extraction returns the talking points: "food" and "wonderful staff."

**Describe named entity recognition**

*Named entity recognition (NER)* is the identification of entities in a piece of text. Entities are of a type, or class (such as people, places, organizations, numbers, or personal information), and are objects that are already known by the AI workload. Some entities like dates have sub-types, such as time or duration.

Following are common entities available with NER:

* People
* Places
* Organizations
* Dates and times
* Date
* Duration
* Time
* Quantities
* Age
* Temperature

NER can also find personal information including email addresses, telephone numbers, social security numbers, driver license numbers, and passport numbers.

NER can be used to extract brand information in emails and social media posts. You can use NER to find mentions of your product names. For example, the Content Moderator service uses NER to scan text against a list of terms that you have provided.

You can use NER to scan news feeds for mentions of people or organizations and tag those documents. For example, you can create a curated list of daily news articles that are relevant to your company or clients.

**Describe sentiment analysis**

A common use for NLP is to score text for sentiment. *Sentiment analysis* is the determination if a piece of text has a positive or negative feeling or emotion.

Sentiment analysis can be used to listen to what customers think of your products and services by analyzing social media for positive or negative sentiment. There are many third-party applications—for example, Hootsuite—that monitor and track social media using sentiment analysis, allowing you to monitor trends and to take action.

**Describe speech recognition and synthesis**

NLP is not just for text; it includes the spoken word. Speech covers both *speech recognition*, detecting and interpreting spoken input and converting into text, and *speech synthesis*, generating spoken output from text.

**Speech recognition**

Speech recognition does not simply recognize words; it must find patterns in the audio. Speech recognition uses an acoustic language model that converts the audio into phonemes. A *phoneme* is the smallest unit of sound in speech. When we teach children to read, we teach how letters are represented by sounds. In English, however, there are 26 letters in the alphabet, but there are 44 phonemes, and similar words can be pronounced very differently.

The baked good, scone, is an example of the different phonemes. Scone has two common pronunciations in England: one that rhymes with “cone” and another that rhymes with “gone.” If you are in Scotland, you might hear it pronounced as “skoon.”

Once the phonemes have been identified, the acoustic model then maps phonemes to words, using statistical algorithms that predict the most probable sequence of words based on the phonemes.

An example of speech recognition can be found in Microsoft PowerPoint. The Presenter Coach tool monitors your speech and uses speech recognition to give you a statistical report for a rehearsal of your presentation. It will tell you if you have used filler words or euphemisms, and it detects if you are just reading the text from the slide. It will also provide suggestions to improve your delivery.

Other common NLP scenarios for speech recognition include interactive voice response in call centers, transcribing telephone calls, and in-home automation.

**Speech synthesis**

Synthesized speech does not generate a sound for each word. It converts the text into phrases, and then using the acoustic model, converts the phrases into smaller prosodic units and then into phonemes.

A voice is then applied to convert the phonemes into audio speech. The voice defines the pitch, speaking rate, and intonation for the generated audio.

You find speech synthesis in personal digital assistants, like Siri and Cortana, that respond vocally to your requests.

Common NLP scenarios for speech synthesis are broadcasting arrivals and departures at airports, reading out text messages while you are driving your car, and screen reading software applications for visually impaired people.

**Describe translation**

There are many languages in the world, and the ability to convert text from one language into another language is a feature of NLP.

Translation is very difficult for humans to do. Disclaimer: My wife is a translator from Italian to English. Languages do not have simple word-for-word translations; for instance, there are words in Italian that can be one of multiple words in English, depending on the context. There are Italian words that do not have a simple translation, such as Ciofeca, which means a poor quality and badly prepared drink, such as coffee. Another example is *volume* in Italian means the space inside an object, whereas *volume* in English generally is a measurement of that space. *Volume* can also refer to the level of sound, or a book in a series. The German word, Schadenfreude, has no direct translation and generally means to derive pleasure from someone else’s misfortune. In Finnish, the word Kalsarikännit means the feeling you get when sitting at home getting drunk in your underwear. As you can see, translation is hard and relies on context and a lot of knowledge.

Most human translators translate from one language into their native language; they do not translate the other way around, as they need to make the text understandable to the native speaker. You have probably all seen instruction manuals poorly translated where a native speaker has not been used.

To translate from one language to another requires models for both languages and to be able to understand the context of how each language is used. This involves understanding rules of grammar, use of informal language, and large dictionaries and glossaries.

AI-powered text translation uses large amounts of translated text to train the translation models. You will find that the translations are better where there are more examples of the text available in each language, so the results of translation between English and other languages are often better than translation between other languages. Translation results tend to be better on news and marketing documents than on highly technical documents, as again, there are more examples of the former available to train the models on.

Translation includes the conversion of both text and audio speech from one language into another:

* **Text**   Text translation translates the text documents from one language to another language.
* **Speech**   Speech translation translates spoken audio from one language to another language.

Now that we have explained the common NLP workloads, we will look at the services in Azure Cognitive Services for the major NLP workloads.

**Skill 4.2: Identify Azure tools and services for NLP workloads**

Azure Cognitive Services provides pre-trained NLP models that cover most of the capabilities required for analyzing text and speech. This section describes the capabilities of the main language services within Azure Cognitive Services.

These language services allow developers to add features such as sentiment detection, speech recognition, and language understanding to their applications without needing skills in machine learning. These language services are available for both text and speech.

The text services extract meaning from unstructured text:

* **Text Analytics**   Discovers insights from textual data. Text Analytics is one of the most heavily used Cognitive Services.
* **Language Understanding service (LUIS)**   Interprets the intent and extracts key information from supplied text. LUIS is used by other Cognitive Services to provide understanding of text.
* **Translator**   Detects and translates text in real time, or in batch, across more than 90 languages.
* **Immersive Reader**   Helps readers of all ages and abilities to comprehend text using audio and visual cues.
* **QnA Maker**   Creates a question-and-answer knowledge base from existing documents and sources. QnA Maker is explained in [Chapter 5](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch05.xhtml#ch05).

The speech services allow you to add speech processing into your apps:

* **Speech to Text**   Transcribes audio into text in real time or from audio files.
* **Text to Speech**   Synthesizes text into spoken audio.
* **Speech Translation**   Converts audio into text and translates into another language in real time. Speech Translation leverages the Translator service.
* **Speaker Recognition**   Identifies people from the voices in an audio clip.

The Microsoft Azure AI Fundamentals certification encompasses the capabilities of Natural Language Processing services. This requires you to understand the capabilities of the services described in this section.

This skill covers how to:

* Identify the capabilities of the Text Analytics service
* Identify the capabilities of the Language Understanding service (LUIS)
* Identify the capabilities of the Speech service
* Identify the capabilities of the Translator service

Image ***EXAM TIP***

You need to be able to map the service to the scenario presented and to identify the operation for a service to use.

Language services are used alongside other Cognitive Services, such as OCR and Content Moderation described in [Chapter 3](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch03.xhtml#ch03), and with bots described in [Chapter 5](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch05.xhtml#ch05).

To use the language services, you will need to create a Cognitive Services multi-service resource, or a language single-service resource, as described in [Chapter 3](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch03.xhtml#ch03).

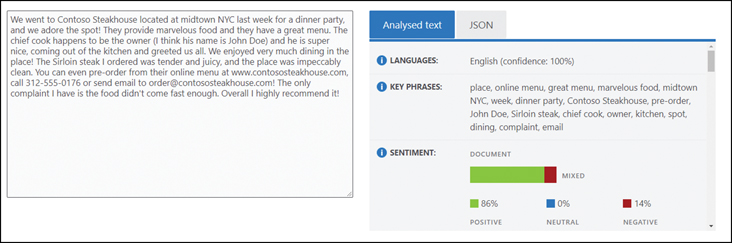
The following sections describe the capabilities of four of the main language APIs provided by Cognitive Services.

**Identify the capabilities of the Text Analytics service**

The Text Analytics service is an Azure Cognitive Service that performs a series of Natural Language Processing operations on text.

Text Analytics can detect the sentiment of sentences or whole paragraphs. You can extract key phrases from a piece of text, and extract entities such as people, places, and things from a piece of text.

You can see how Text Analytics works without an Azure subscription at <https://azure.microsoft.com/services/cognitive-services/text-analytics/>, as shown in [Figure 4-2](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig02). The text for a restaurant review is analyzed, and the results are presented both in a friendly format and in JSON format.



**FIGURE 4-2** Text Analytics

The Text Analytics service can be deployed in the Azure portal by searching for Text Analytics when creating a new resource. You must select your region, select the resource group, provide a unique name, and select the pricing tier: Free F0 or Standard.

You can create Text Analytics resources using the CLI, as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-124pro01a)

az cognitiveservices account create --name <unique name> --resource-group <resource

group name> --kind TextAnalytics --sku F0 --location <region>

The following sections describe the capabilities of the operations available with the Text Analytics service.

**Detect language**

Text Analytics supports a wide range of languages. A key operation is the identification of the language used in a document. Text Analytics can detect a wide range of languages, variants, dialects, and some regional/cultural languages—for example, French (fr-Fr) vs. Canadian French (fr-CA).

The Detect language operation returns the name of the language, the ISO language code, and a level of confidence between 0 and 1. If there is more than one language in the document, then the predominant language is returned.

The request URL is formulated as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-124pro02a)

https://{endpoint}/text/analytics/v3.0/languages

You supply a JSON document containing text to be analyzed. Following is the JSON returned for the paragraph on translation from earlier in the chapter that has a mixture of ­English, German, Italian, and Finnish words:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-124pro03a)

"documents": [

{ "id": "1", "detectedLanguage":

{ "name": "English", "iso6391Name": "en", "confidenceScore": 1.0 },

This operation found that the language was English, an ISO code “en,” with a confidence score of 100%.

**Sentiment analysis**

Sentiment analysis analyzes the emotion for each sentence in a piece of text and for the whole document.

Sentiment is a classification model that evaluates the emotion of the text as to how positive or negative it is. The operation returns a sentiment score between 0 and 1, with 1 as the most positive, and a sentiment label (positive, negative, neutral, or mixed).

Sentiment analysis uses a model that has been pre-trained on millions of examples of text. Currently, sentiment can be evaluated for 13 languages.

The request URL is formulated as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-125pro01a)

https://{endpoint}/text/analytics/v3.0/sentiment

You supply a JSON document containing text to be analyzed. Following is the JSON returned for parts of the paragraph on translation from earlier in this chapter:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-125pro02a)

"documents": [{ "id": "1",

"sentiment": "mixed", "confidenceScores": { "positive": 0.31, "neutral": 0.01,

"negative": 0.68 },

"sentences": [

{ "sentiment": "negative", "confidenceScores": { "positive": 0.0, "neutral": 0.0,

"negative": 1.0 }, "offset": 0, "length": 47, "text": "Translation is very difficult for

humans to do." },

{ "sentiment": "negative", "confidenceScores": { "positive": 0.0, "neutral": 0.0,

"negative": 1.0 }, "offset": 48, "length": 65, "text": "Translation is hard and relies

on context and a lot of knowledge." },

{ "sentiment": "positive", "confidenceScores": { "positive": 0.93, "neutral": 0.03,

"negative": 0.04 }, "offset": 114, "length": 37, "text": "Luckily, Cognitive Services

can help" }]

This operation found that the overall sentiment of the paragraph is mixed with a positive score of 31% and a negative score of 68%. The first and second sentences are 100% negative, while the third sentence is 93% positive.

**Key phrase extraction**

Key phrase extraction identifies the main talking points in unstructured text. The operation generates a list of relevant words and/or phrases that describes the subject of the document. It should be noted that the phrases are returned as a simple list without any context.

Key phrase extraction uses a model that has been pre-trained on millions of examples of text. Currently, phrases can be extracted for 16 languages.

The request URL is formulated as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-125pro03a)

https://{endpoint}/text/analytics/v3.0/keyPhrases

You supply a JSON document containing text to be analyzed. Following is the JSON returned for parts of the paragraph on translation from earlier in the chapter:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-125pro04a)

"documents": [{ "id": "1", "keyPhrases": ["Translation", "context", "lot of knowledge",

"humans", "Cognitive Services"]}

The key phrase extraction operation found five key phrases in the paragraph of text.

**Named entity recognition**

Named entity recognition extracts a wide set of prebuilt entities from documents. Entities are objects in the text such as people, places, organizations, date/time, numbers, and personally identifiable information (PII).

***NOTE*   ENTITY RECOGNITION**

Microsoft has referred to named entity recognition as entity recognition in the past but is now using the more common term in its documentation. In the exam, you may see usage of entity recognition. Named entity recognition and entity recognition refer to the same operation.

The categories of entities extracted by Text Analytics are:

* **Person**   The names of people. This is not limited to famous people but can identify forenames and surnames in the text.
* **PersonType**   The job type or job title.
* **DateTime**   Dates and times of day including durations and date ranges.
* **Quantity**   Numerical measurements and units including temperature, percentages, ages, and dimensions.
* **Location**   A geographical feature, landmark, building, or city.
* **Organization**   The names of companies, political groups, musical bands, sports teams, government bodies, and public organizations.
* **Event**   The names of historical, social, and other events.
* **Product**   Physical objects. Currently these are computing related.
* **Skill**   Capabilities, skills, or expertise.
* **Address**   Addresses including street, city, and postal code.
* **Phone number**   Telephone numbers.
* **Email**   Email addresses.
* **URL**   URLs to websites.
* **IP**   Network IP addresses.

The operation returns the entity, its category, and a confidence score between 0 and 1. ­Currently, entities can be extracted for 23 languages, including Arabic.

***NOTE*   ENTITY LINKING**

Text Analytics also provides links to Wikipedia articles for well-known entities.

The request URL is formulated as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-126pro01a)

https://{endpoint}/text/analytics/v3.0/entities/recognition/general

You supply a JSON document containing text to be analyzed. Following is the JSON returned for the parts of the paragraph on translation earlier in the chapter:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-127pro01a)

"documents": [{ "id": "1", "entities": [

{ "text": "Translation", "category": "Skill", "offset": 0, "length": 11,

"confidenceScore": 0.8 },

{ "text": "Azure", "category": "Organization", "subcategory": "Sports", "offset": 123,

"length": 5, "confidenceScore": 0.53 }]

Two entities were extracted: Translation as a skill with confidence score of 80% and Azure as a sport organization with a confidence score of 53%. Clearly, this second entity is incorrect.

Linked entities were also identified.

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-127pro02a)

"documents": [{ "id": "1", "entities": [

{ "name": "Translation", "matches": [{ "text": "Translation", "offset": 0, "length":

11, "confidenceScore": 0.13 }, { "text": "Translation", "offset": 48, "length": 11,

"confidenceScore": 0.13 }], "language": "en", "id": "Translation", "url": "https://

en.wikipedia.org/wiki/Translation", "dataSource": "Wikipedia" },

{ "name": "Cognitive computing", "matches": [{ "text": "Cognitive Services", "offset":

123, "length": 18, "confidenceScore": 0.76 }], "language": "en", "id": "Cognitive

computing", "url": "https://en.wikipedia.org/wiki/Cognitive\_computing", "dataSource":

"Wikipedia" }]

Two Wikipedia articles have been identified: one for Translation and the other related to Cognitive Services.

**Use cases for Text Analytics**

I have implemented solutions for call centers for much of my career using a variety of tools. Call centers require feedback as they need to measure and improve customer satisfaction and evaluate call center agent performance.

Many call centers use feedback tools such as post call surveys and listening to calls manually, which can be both inaccurate and time consuming. We can use Text Analytics to aid staff and managers to gain statistics and insights into telephone calls, as described next:

* First, Speech to Text transcribes call recordings recorded calls to text.
* Voice recognition can be used to break the text into who is speaking in turn.
* Sentiment analysis can analyze the sentiment of each interaction. We can then track:
  + How the customer sentiment changes over the lifetime of the call.
  + If the agent can convert negative exchanges into positive conversations.
* Key phrase extraction can extract the main talking points in the conversation. This can be used to categorize the call.
* Named entity recognition can extract entities such as people’s names, company names, locations, dates, and personal information. This data can enhance the data held for the customer.

Another use case for Text Analytics is in handling compliance. You can scan the emails and call recordings made by your sales team to automate compliance checking by scanning for mentions of key phrases or named entities that represent your products and services.

**Identify the capabilities of the Language Understanding service (LUIS)**

The Language Understanding service (LUIS) is a key component of Azure Cognitive Services and is used by many of the other Cognitive Services. LUIS is a Natural Language Understanding (NLU) service that analyzes user input and uses a language model to understand the meaning of that text input.

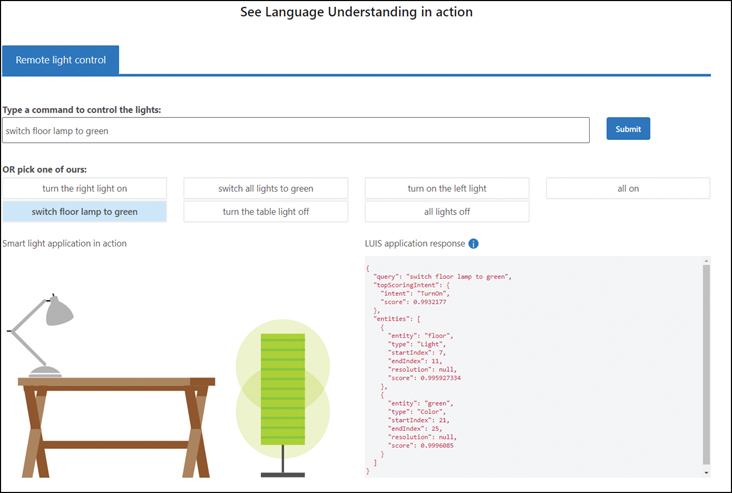
A major strength of AI-based solutions is the ability to converse with users using natural language. Chatbots are a common tool that requires language understanding and is one of the major consumers of LUIS. Using LUIS, you can add conversational intelligence into your bot. We will look further at conversational AI in the next chapter.

Fully understanding what a conversation is about is a hard AI problem to solve. LUIS only focuses on identifying the user’s intention and extracting information from a small piece of text. LUIS attempts to identify:

* What the user wants.
* What the user is talking about.

LUIS uses language modeling algorithms developed by Microsoft over many years to apply intelligence to a user’s natural language text. LUIS uses the techniques outlined earlier in the chapter together with the language model to predict the user’s overall meaning and to extract relevant information. Your applications can then use this information to respond intelligently to the user or to carry out the user’s instructions.

You can see how LUIS works without an Azure subscription at <https://azure.microsoft.com/services/cognitive-services/language-understanding-intelligent-service/#demo>, as shown in [Figure 4-3](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig03). In this demo, you can either type an instruction or pick an example. When you click on Submit, the instruction is evaluated by LUIS, and the action taken is shown in the image.



**FIGURE 4-3** LUIS demo

The phrase “switch floor lamp to green” has been evaluated, and LUIS has identified that the request is to “TurnOn” the “light” on the “floor” and set the “color” to “green.”

LUIS is trained to handle common business domains, including customer service, restaurant reservations, and home automation. Unlike the other language services that are pre-trained, you can create models for your own domains if the prebuilt options do not meet your business scenario.

**Key concepts**

There are three key concepts in LUIS that you need to understand before creating LUIS applications, as follows:

* **Utterances**   An *utterance* is an example phrase that a user says. You need to specify sample utterances to train your model.
* **Intents**   An *intent* is the action that the user wants to perform. An intent is linked to multiple utterances.
* **Entities**   Entities are like the entities in NER. *Entities* are the subjects or context for the intent.

Image ***EXAM TIP***

Understanding the difference between these concepts is a fundamental skill in this exam.

You first define your intents. Intents are linked to the actions that your client application can perform. You should create an intent when you want to trigger an action in your client application. You then add a few potential utterances to each intent. LUIS then takes these examples of phrases with the intents and starts training the model. These training utterances are used by your LUIS model to determine which intent the user is referring to.

Let’s look at an example for intents and utterances. We will use a LUIS model to power a bot that handles requests around creating and using Cognitive Services. Consider this list of intents and associated utterances:

* CreateCognitiveService
  + “I must deploy Cognitive Services”
  + “I want to create a Cognitive Service resource”
  + “I need to generate a new LUIS authoring resource”
  + “Create Azure Cognitive Services”
* CallCognitiveService
  + “I want to evaluate the sentiment for this sentence”
  + “I need to determine which language this text”
  + “I have to translate this document”
  + “What is the best service for extracting text from an image”

LUIS does not have deep sematic Natural Language Processing. For instance, LUIS cannot automatically differentiate different verb tenses or alternative words. As an example, LUIS is unable to determine if the words add, adding, and added have the same intent. LUIS is unable to automatically recognize that the words add, create, and generate have the same intent. Therefore, you need to provide a set of utterances to help LUIS handle the different ways a user might phrase their requests. LUIS has been developed so that you only need to provide a few sample utterances for LUIS to create a model with good accuracy and do not have to provide every possible variation yourself.

When creating utterances, you need to provide different ways of saying the same thing with different verb tenses and substitute wording. Microsoft recommends that you should create between 10 and 30 utterances per intent. The more example utterances you provide, the more accurate your model will be.

You also need intents and utterances for greetings and other non-action phrases that a user might employ.

Entities are the information required to perform the action behind the intent; they are the data for the action. For the previous example, the entities might be:

* Cognitive Services
* LUIS authoring resource
* Resource
* Document
* Sentence
* Text
* Paragraph
* Image

Once you define your utterance and entities, you can improve the accuracy of the language model by adding hints, known as *features*, by providing variations for the words used. LUIS will then use these features when recognizing the intent and entities.

There are prebuilt entities you can use in your model, and you can specify your own custom entities. There are four types of entity you can create:

* **Machine learned**   Entities that are learned from your utterances. When you train the model, the entities are identified from the labels you apply to the utterances.
* **List**   A simple list of items for exact text matching. You can also supply synonyms. For the example, you could add “Document” to a list with the synonyms “paragraph,” ­“sentence,” and “text.”
* **Regex**   A regular expression—for example, telephone numbers or postal codes—that matches exactly.
* **Pattern**   Pattern.any is used where utterances are very similar but refer to different entities. For example, you can use the pattern {Azure service} to extract “Language Understanding” from the utterance “How should I create a Language Understanding resource in the Azure portal.”

***NOTE*   INTENTS VS. UTTERANCES VS. ENTITIES**

An intent is the required outcome from an utterance and is linked to an action. Entities are data in an utterance. Entities are the information needed to perform the action identified in the intent.

**Prebuilt models**

LUIS contains several prebuilt models. These models provide combinations of intents, utterances, and entities. You can use a prebuilt domain model that contains intents, utterances, and entities, or you can use a prebuilt intent model that contains intents and utterances, or you can use a prebuilt entity model.

The following prebuilt domain models are available in LUIS:

* **Calendar**   Making appointments.
* **Communication**   Messaging and telephone calls.
* **Email**   Reading and replying to emails
* **HomeAutomation**   Controlling smart devices.
* **Notes**   Taking notes.
* **Places**   Organizations, restaurants, and public spaces.
* **RestaurantReservation**   Booking tables in a restaurant.
* **ToDo**   Managing tasks.
* **Utilities**   Common tasks that you can use in any domain.
* **Weather**   Weather forecasts.
* **Web**   Navigation to websites.

Prebuilt intent models contain intents and utterances but not entities. The intents from the prebuilt domains can also be added without adding the whole domain model.

Prebuilt entities handle many of the common concepts you need for LUIS apps. The following prebuilt entities are available:

* Age
* Currency
* DateTime
* Dimension
* Email
* Geography
* KeyPhrase
* Number
* Ordinal
* Percentage
* PersonName
* Phonenumber
* Temperature
* URL

**Custom schema**

A *custom schema* consists of intents and, optionally, entities. A new custom schema has no intents or models. You can add any prebuilt domain, intents, and entities to a custom model. You are not just restricted to a single prebuilt model but can add as many as required.

You can, of course, create your own intents, utterances, and entities and combine them with the prebuilt models to create the schema for your LUIS app.

**LUIS app**

To use LUIS, you will need to create a LUIS app that describes the model for your domain. LUIS requires both an authoring resource and a prediction resource. The authoring resource is used to create, manage, train, test, and publish your applications. The prediction resource is used by other applications after you publish your LUIS application to understand text inputs.

LUIS uses a web portal (https://www.luis.ai) where you can create your model, add example utterances, train the model, and finally deploy the app.

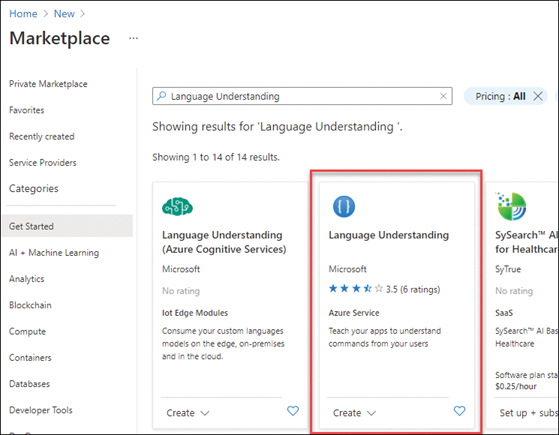
Understanding machine learning is not necessary to use LUIS. Instead, you define the intents and entities and then provide example utterances to LUIS and relate how those utterances are related to intents and entities. LUIS uses this information to train the model. You can improve the model interactively by identifying and correcting prediction errors.

The process for creating a LUIS app is as follows:

* **Build a LUIS schema**   Define the domain and add intents and entities.
* **Add utterances**   Add training example phrases for each intent.
* **Label entities**   Tag the entities in each utterance.
* **Add features**   Create phrase lists for words with similar meanings.
* **Train**   Train the model in the app.
* **Publish**   Publish the app to an endpoint using the prediction resource.
* **Test**   Test your LUIS app using the published endpoint.

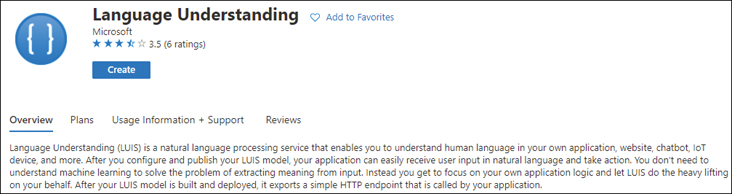
Before you can create a LUIS app, you need to create your LUIS resources. You need both an authoring resource and a prediction resource. The rest of this section will walk you through creating LUIS resources and creating a LUIS app.

To create a resource for LUIS resource in the Azure portal, search for Language Understanding and pick the service titled just Language Understanding, as shown in [Figure 4-4](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig04).



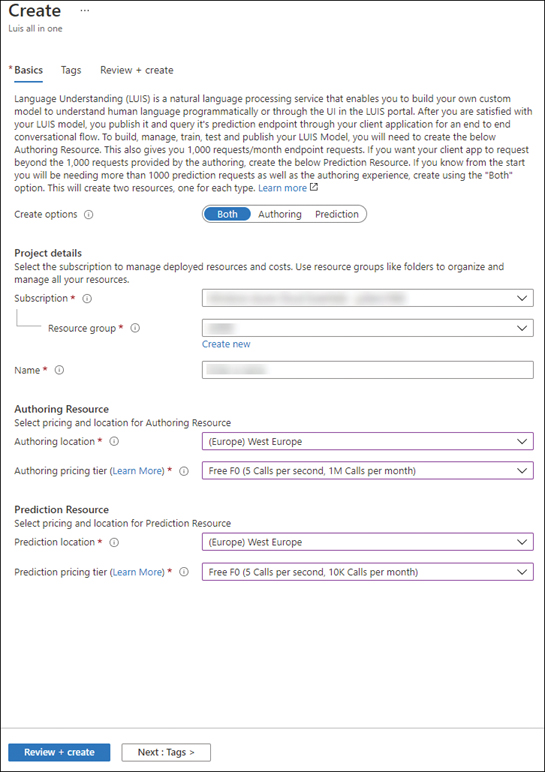
**FIGURE 4-4** Language services in Azure Marketplace

Clicking on Create will show the description for the service, as shown in [Figure 4-5](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig05).



**FIGURE 4-5** Service description for the LUIS service

After clicking on the Create button, the Create Language Understanding (LUIS) pane opens, as shown in [Figure 4-6](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig06).



**FIGURE 4-6** Creating LUIS resources

There is a toggle to choose which service(s) you require: Authoring and/or Prediction. You will need to select your subscription and resource group. You will then need to create a unique name for the service. This name will be the domain name for your endpoints and so must be unique worldwide. For the authoring resource, you should select the region where the authoring resource is to be deployed and select your pricing tier; Free F0 is the only option. You then need to select the region and pricing tier for the prediction resource. The pricing tier for the prediction resource can either be Free F0 or Standard S0.

Clicking on Review + Create will validate the options. You then click on Create to create the resource. If you selected Both, two resources will be deployed with the authoring resource using the name you provided appended with “-Authoring” and the prediction resource using the name you provided.

You can create the LUIS resources using the CLI as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-135pro01a)

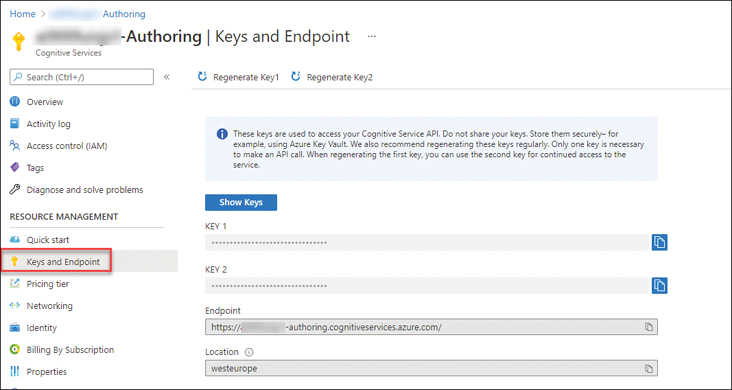
az cognitiveservices account create --name <unique name for authoring> --resource-group

<resource group name> --kind LUIS.Authoring --sku F0 --location <region>

az cognitiveservices account create --name <unique name for prediction> --resource-group

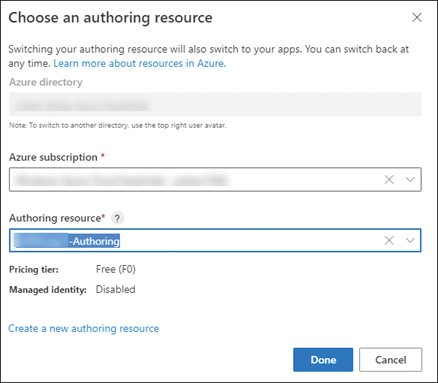
<resource group name> --kind LUIS --sku F0 --location <region>

Once your resources have been created, you will need to obtain the REST API URL and the key to access the resources. To view the endpoint and keys in the Azure portal, navigate to the resource and click on Keys and Endpoint, as shown in [Figure 4-7](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig07).



**FIGURE 4-7** Keys and Endpoint

Once you have created your LUIS resources, you can open the LUIS web portal (https://www.luis.ai) in a browser. You will be prompted to select your subscription and authoring resource, as shown in [Figure 4-8](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig08).



**FIGURE 4-8** Selecting the LUIS authoring resource

You then need to click on + New app in the LUIS portal to create a new app. You need to enter a name for the app, select the culture, select the language the app will understand, and choose your prediction resource. If a window appears titled, “How to create an effective LUIS app,” you can close this window.

***NOTE*   LUIS RESOURCES**

There are two resources in a LUIS app: the authoring resource used to build, manage, train, test, and publish your LUIS model, and a prediction resource to query the model. You can use the multi-service Cognitive Services resource for prediction, but you must create an authoring resource in order to create a LUIS app.

You then need to click on + New app in the LUIS portal to create a new app. You must enter a name for the app, select the culture, the language the app will understand, and choose your prediction resource. You will then see the authoring window for your app, as shown in [Figure 4-9](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig09).



**FIGURE 4-9** Authoring LUIS app

Next, you should add prebuilt domains. Click on Prebuilt Domains in the left-hand navigation pane, select the following domains, and click on Add domain:

* Utilities
* Web

Click on Intents in the left-hand navigation pane. You will now see that a number of intents from these domains have been added to your app.

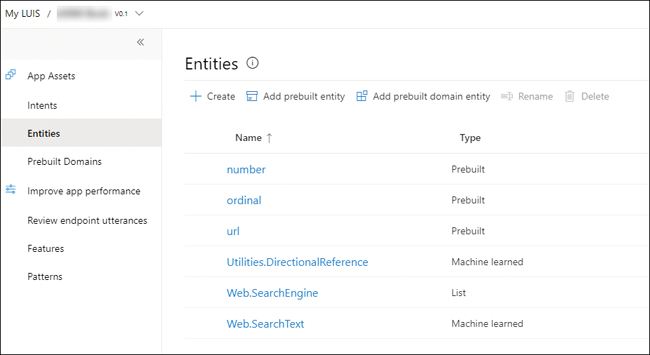
You can now add your own intents. Click + Create, create the following intents, and add the example user inputs (utterances).

* CreateCognitiveService
* CallCognitiveService

For each intent, add example user inputs, with utterances such as the following:

* “I must deploy Cognitive Services”
* “I want to create a Cognitive Service resource”
* “I need to generate a new LUIS authoring resource”
* “Create Azure Cognitive Services”
* “I want to evaluate the sentiment for this sentence”
* “I need to determine which language this text”
* “I have to translate this document”
* “What is the best service for extracting text from an image”

Click on Entities in the left-hand navigation pane. You will see the prebuilt entities already added from the prebuilt domains you selected, as shown in [Figure 4-10](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig10).



**FIGURE 4-10** LUIS entities

Click on the Add prebuilt entity and add the following entities:

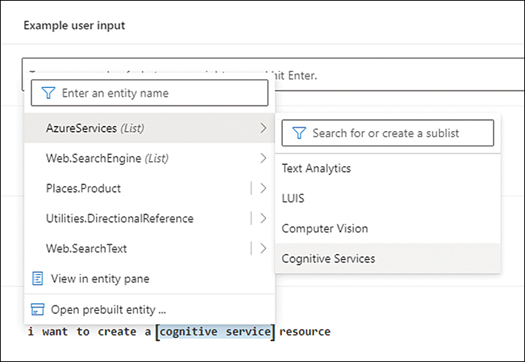
* keyPhrase
* personName

Click on the Add prebuilt domain entity and add the Places.product entity.

You can now add your own entities. Click + Create, name your entity AzureServices, and choose the List type. Enter the following values and synonyms:

* Cognitive Services
* Computer Vision
* Custom Vision
* Face
* OCR
* LUIS
* Language Understanding
* Text Analytics
* Sentiment Analysis
* Language Detection
* Translator

You now need to tag the entities in each of the utterances. Edit the CreateCognitiveService intent. Use the mouse to select the name of an Azure service in an utterance. As you select a word or phrase, a window will appear where you can select an entity. Choose the AzureServices entity, as shown in [Figure 4-11](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig11).



**FIGURE 4-11** Tag an utterance with an entity

Edit the CallCognitiveService intent and click on + Add feature. Click on + Create new phrase list. The phrase list should be named TextSources with these values: Document, Paragraph, Sentence, and Text.

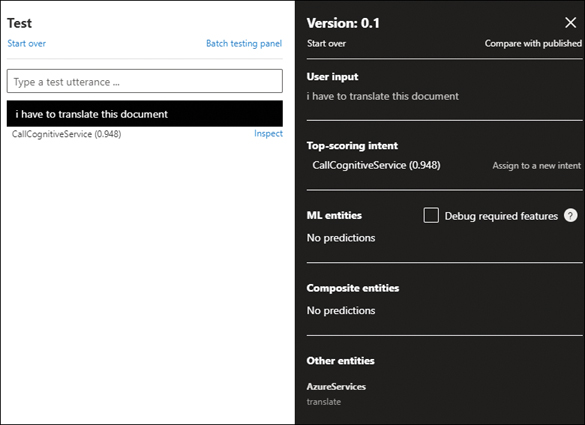
The intent will now look similar to [Figure 4-12](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig12).



**FIGURE 4-12** Intent with tags and features

You are now ready to train your model. Click on the Train button at the top of the LUIS authoring page. Training will take a few minutes.

The LUIS portal allows you to test your app interactively. Click on the Test button at the top of the LUIS authoring page. A pane will appear where you can enter a test utterance. First try one of the example utterances you add to an intent—for example, “I have to translate this document.” You can see the results as shown in [Figure 4-13](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig13).



**FIGURE 4-13** Test results

You should see in the results the selected intent (action) and entity (data). This is the information that your client application can use to perform the action on the data.

Try other phrases with different wording and evaluate the model. If LUIS does not correctly predict the intent, you can fix this by assigning to the correct intent.

When you have completed the testing of your LUIS app, you can publish the app. After publishing a prediction endpoint URL will be available.

To use a Language Understanding (LUIS) model to find the intent of a text statement in a client application, you need the ID of the LUIS app and the endpoint and key for the prediction resource, not the authoring resource.

**Use cases for LUIS**

LUIS is not typically used on its own. It is used to provide enhanced user experience in other applications.

For instance, you could create an application that provides a searchable interface for a medical database and document store. You could create a LUIS app with entities for specific medical terminology to help the user find the relevant entries in the database.

Another use is in gaming, where you have the player either issuing verbal or written instructions, in single- and multi-player modes. For example, there a lot of different ways for a player to say, “let’s go,” especially in stressful situations or when reacting instinctively to the appearance of danger. There is “Get me out of here,” “Beam me up,” “Engage the warp-drive,” and so on.

LUIS can be used with IoT devices to turn a user’s command into an action that is understandable by the device.

We will see in the next chapter how LUIS is used with QnA Maker and chatbots.

**Identify the capabilities of the Speech service**

Speech recognition has long been a goal for AI. After over 15 years of development, it is only recently that algorithms have been developed that have enabled a set of Cognitive Services to be made available for you to use in your applications as a set of easy-to-consume APIs. These APIs are the same ones that Microsoft uses in its own products, such as PowerPoint’s live captions and subtitles.

Modern speech recognition technology relies on machine-learned statistical models that leverage cloud computing combined with vast quantities of sample audio. Microsoft’s speech recognition technology handles dialects and accents of different speakers; it can deal with jargon, and it performs well in noisy conditions.

The Speech service contains an acoustic model and a language model for many languages. The acoustic model in the Speech service is a deep neural network trained on thousands of hours of audio using advanced algorithms. The number of languages varies by operation and is increasing all the time.

The Speech service can be deployed in the Azure portal by searching for Speech when creating a new resource. You must select your region, select the resource group, provide a unique name, and select the pricing tier: Free F0 or Standard S0. Be careful, and do not create the service called Speech to Text; this is a third-party paid-for service.

You can create Speech resources using the CLI, as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-141pro01a)

az cognitiveservices account create --name <unique name> --resource-group <resource

group name> --kind SpeechServices --sku F0 --location <region>

The Speech service has several speech-processing operations.

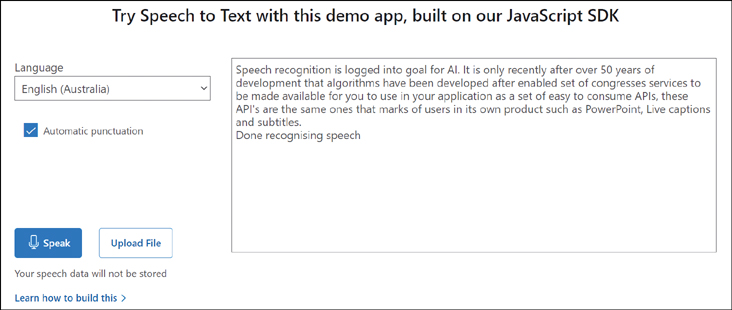
**Speech to Text**

The Speech to Text API detects and transcribes spoken input into text. The Speech to Text operation can transcribe audio into text in real-time or from a recording. It converts fragments of sound into text using the acoustic model and then uses the language model to create words and phrases. Speech to Text converts audio from a range of sources, including microphones, audio files, and Azure Blob storage.

The Speech to Text API can be used synchronously (real-time) or asynchronously (batch). There are two separate APIs: one for short audio (up to 60 seconds) that you can transcribe in real-time, and the other for batch transcription. The batch Speech to Text API can translate large volumes of speech audio recordings stored in Azure Blob Storage.

More than 85 languages and variants are supported.

You can see how Speech to Text works without an Azure subscription at <https://azure.microsoft.com/services/cognitive-services/speech-to-text>, as shown in [Figure 4-14](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig14).



**FIGURE 4-14** Speech to Text

In this example, the first paragraph from this section was read aloud using the computer’s microphone. The speech was recognized and transcribed reasonably accurately but has a couple of errors. It could not differentiate between 15 and 50, and mis-transcribed Cognitive as “congresses.” The Speech to Text service is typically faster and more accurate than a human being can achieve.

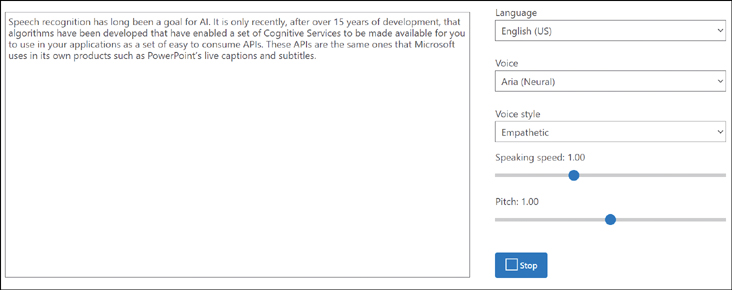
Speech to Text partitions the audio based on the speakers’ voices to determine who said what. This allows you to obtain transcripts with automatic formatting and punctuation.

**Text to Speech**

Text to Speech is useful when you cannot look at a screen if you are controlling other equipment or are using a mobile device. Text to Speech generates, or synthesizes, text into spoken audio.

The Text to Speech API converts text into synthesized speech. You can choose from neutral voices, standard voices, or a custom voice. You can also create your own custom voice for use in speech synthesis. There are over 200 voices available, and the Text to Speech service supports over 60 languages and variants.

You can see how Text to Speech works without an Azure subscription at <https://azure.microsoft.com/services/cognitive-services/text-to-speech>, as shown in [Figure 4-15](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig15).



**FIGURE 4-15** Text to Speech

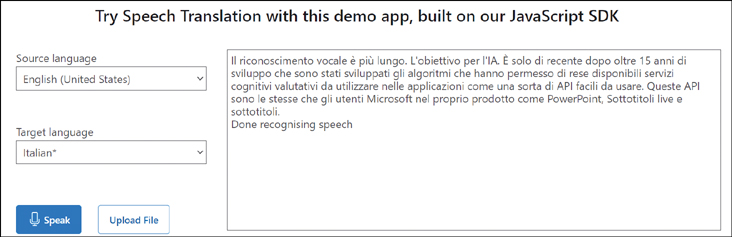
In this example, the first paragraph from this section is synthesized into audio and played through the computer’s speakers. The results are very impressive and very lifelike.

**Speech Translation**

You could achieve translation of speech yourself using a mixture of the Speech to Text, Translator, and Text to Speech services. The Speech Translation service simplifies this process for you. It first detects and transcribes speech into text; then it processes to make it easier for translation. The text is then fed to the text translation service to convert to the target language. Finally, the translated text is synthesized into audio.

Speech Translation converts audio into text and translates into another language in real-time or in batch. Speech Translation can translate audio into more than 60 languages. Speech Translation performs both speech-to-text and speech-to-speech translations.

You can see how Speech Translation works without an Azure subscription at <https://azure.microsoft.com/services/cognitive-services/speech-translation>, as shown in [Figure 4-16](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04.xhtml#ch04fig16).



**FIGURE 4-16** Speech Translation

In this example, the first paragraph from this section was read aloud using the computer’s microphone. The speech was recognized and translated into the target language. However, it has not translated it correctly; the first half of the text has been split incorrectly into sentences and has several errors. The second half is reasonably well translated.

**Speaker Recognition**

Speaker Recognition identifies speakers from their voice characteristics in an audio clip. Speaker Recognition answers the question: “Who is speaking?”

To be identified, speakers must be enrolled using sample audio recordings of their voice. The Speaker Recognition service extracts the characteristics of the voice to form a profile. This profile is used to verify that the speaker is the same person (speaker verification) or to identify the speakers in a conversation (speaker diarization). Speaker diarization can be used to assist in creating transcripts of conversations.

**Use cases for Speech**

Mobile phone apps are a common use case for speech-related services. Many apps now use speech to text, text to speech, and speaker recognition. There are apps such as Microsoft Translator that work on your mobile and translate speech across two languages.

Personal Digital Assistants such as Cortana use the Speech service to take instructions and questions as audio through the microphone and respond with audio. You will find speech services in home automation.

My banking app allows me to use my voice to identify me. I had to enroll my voice in the app by saying the statement “My identity is secure because my voice is my passport, verify me” three times. I can now access my banking details by repeating the same phrase.

Telephony data that is generated through landlines and mobile phones is typically low quality, which creates challenges when converting speech into text. Recent improvements in the Speech service have significantly increased the accuracy of transcribing telephone call recordings.

**Identify the capabilities of the Translator service**

The Translator service translates text from one source language (from) into multiple other (to) languages. The Translator service allows you to specify multiple “to” languages, so you can simultaneously translate into multiple languages.

The Translator service uses Neural Machine Translation (NMT). NMT uses neural networks and deep learning to translate whole sentences. Statistical Machine Translation (SMT) technology uses statistical analysis to estimate the best possible translations for a word given in the context of a few neighboring words. Translator replaced SMT with NMT in 2016.

The Translator service can translate text in real-time or batch across 90 languages, variants, and dialects.

**Create a Translator resource**

Translator is a non-regional service, and you should create your resource in the Global region, unless your application requires a specific region.

The Translator service can be deployed in the Azure portal by searching for Translator when creating a new resource. You must select your region, select the resource group, provide a unique name, and select the pricing tier: Free F0 or tiers with volume discounts.

You can create a Translator resource using the CLI as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-144pro01a)

az cognitiveservices account create --name <unique name> --resource-group <resource

group name> --kind TextTranslation --sku F0 --location Global

**Use the Translator service**

Translator is a modern JSON-based Web API. There are several operations that can be performed:

* **Language**   Returns the list of languages supported by the service.
* **Detect**   Identifies the language of the source text.
* **Translate**   Translation of text from one language into multiple languages.
* **Transliteration**   Translation of text from one language into another language and changing the script/character set.
* **BreakSentence**   Identifies the position of the sentences in the text.
* **Dictionary Lookup**   Provides alternative translations of a word or phrase.
* **Dictionary Example**   Provides examples of how a word or phrase is used.

The request URL for the translate operation is formulated as follows:

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-144pro02a)

https://api.cognitive.microsofttranslator.com/translate?api-version=3.0&to<language

code>

The Translator service supports translation into language variants such as French and Canadian French. The Translator service allows you to specify a cultural variant when translating into a language. You append the cultural code to the language code—for example, fr-CA for Canadian French—to the URL request.

Following is the JSON returned for a paragraph at the beginning of this section translated from English (en) to Spanish (es):

[Click here to view code image](https://learning.oreilly.com/library/view/exam-ref-ai-900/9780137358076/ch04_images.xhtml#a-145pro01a)

[{detectedLanguage: {language: "en", score: 1.0},

translations: [{to: "es", text: "El servicio Traductor puede traducir simultáneamente

de un idioma a varios otros idiomas. El servicio Traductor traduce texto de un idioma a

otro idioma. El servicio Traductor le permite especificar varios idiomas, para que pueda

traducir simultáneamente a varios idiomas."}]}]

There are limits in the Translator API:

* Translate operation is limited to 10,000 characters, including spaces.
* Transliterate operation is limited to 5,000 characters.
* Detect operation is limited to 50,000 characters.
* Dictionary Lookup operation is limited to 1,000 characters.

**Use cases for Translator**

Earlier in this section, we gave an example of a mobile app that translates from one language into another. There are many such apps that make life easier for the traveler.

There are other use cases for translation support, such as translating speech and text in multi-country call centers.

You could create a multi-language website using Translator or build an app that supports multiple languages.

Microsoft PowerPoint provides real-time translation of presentation as subtitles. The Live Presentations feature allows audience members to see the presentation on their mobile device with subtitles generated from your speech translated into their language.

**Chapter summary**

In this chapter, you learned some of the general concepts related to Natural Language Processing. You learned about the features of Natural Language Processing, and you learned about the services in Azure Cognitive Services related to language and speech. Here are the key concepts from this chapter:

* Natural Language Processing is about extracting information and insights from both speech and text.
* NLP use statistical analysis and other processes to extract meaning from text.
* NLP uses a language model to understand the text.
* Key phrase extraction finds the main topics in the text.
* Named entity recognition identifies known entities (people, places, and things) in the text.
* Sentiment analysis classifies the emotion in the text as positive or negative.
* Speech recognition detects and interprets audio and converts it to text.
* Speech synthesis generates spoken audio from text.
* Translation converts text from one language into another language.
* The Text Analytics service is an Azure Cognitive Service that performs a series of Natural Language Processing operations on text, including detection of language, sentiment analysis, key phrase extraction, and named entity recognition.
* The Language Understanding service (LUIS) is an Azure Cognitive Service that analyzes user input to understand the meaning of the input text.
* LUIS extracts from the text what the user wants and what the user is talking about using intents, utterances, and entities.
* An intent is the action that the user wants to perform.
* An utterance is an example phrase that a user says. You need to supply sample ­utterances to train your model.
* Entities are the subjects or context for the intent.
* There are prebuilt entities for common objects, such as dates, places, and personal information (PII). You can add your own custom entities for your business domain.
* Features provide hints for LUIS to use to find intent and entities.
* A LUIS app requires both an authoring resource and a prediction resource to be ­provisioned in Azure.
* Speech to Text transcribes audio into text in real-time or from audio files.
* Speech to Text can transcribe large quantities of audio recordings stored in Azure Blob Storage.
* Text to Speech synthesizes text into spoken audio.
* You can create a custom voice for conversion of text into synthesized speech.
* Speech Translation converts audio into text and translates into another language in real-time.
* Speaker Recognition identifies enrolled people from their voices in an audio clip.
* Translator translates text in real-time or in batch.
* Translator supports translation into many languages and variants.

***NEED MORE REVIEW?*   HANDS-ON LABS**

For more hands-on experience with Natural Language Processing, complete labs 7 to 10 at <https://github.com/MicrosoftLearning/mslearn-ai900>.

**Thought experiment**

Let’s apply what you have learned in this chapter. In this thought experiment, demonstrate your skills and knowledge of the topics covered in this chapter. You can find the answers in the section that follows.

You work for Litware, Inc., a company with several brands that supplies business to business services across the world. Litware is interested in analyzing the large amount of text involved in their business using AI.

Litware wants to evaluate how Cognitive Services can improve their internal document categorization.

Litware wants to create a single support desk to handle their worldwide customer base. This central desk will provide consistent responses to customers no matter their location or language.

Litware needs to understand how customers will respond to this move to a single support desk. Customers are sent a questionnaire to ask them about this move. The questionnaire has a series of questions and includes a space for the customer to write their thoughts on this move. Customers can also make a request as part of the questionnaire for more information or for someone to contact them.

As part of this planned move, Litware monitors social media for mentions about these proposed changes and records telephone calls into the existing support desks.

Answer the following questions:

1. Which workload is used to evaluate how the customer feels about the move to a central support desk?
2. Which workload is used to discover the topics mentioned by customers in the questionnaire?
3. Named entity recognition extracts the intent and action from the request in the ­questionnaire. Is this correct?
4. Which workload is used to monitor social media for negative mentions of Litware’s brands?
5. Which workload is used to transcribe telephone calls into the support desk?
6. Which Cognitive Service would you use to mine customer perceptions of Litware’s ­products and services?
7. What examples of how users phrase their requests do you need to provide to the LUIS app?
8. For what information do you need to use a published Language Understanding model to find the meaning in a text statement?
9. Which service do you use to translate the large volumes of telephone calls?
10. Do you have to use a standard voice, or can you create a custom voice for text to speech?

**Thought experiment answers**

This section contains the solutions to the thought experiment. Each answer explains why the answer choice is correct.

1. The emotion expressed in text is an example of the sentiment analysis workload. ­Sentiment analysis evaluates a piece of text and determines if the text has a positive or negative feeling/emotion.
2. The topics, or main talking points, contained in text is an example of the key phrase extraction workload. Key phrase extraction evaluates the text and identifies the key ­talking points in the text.
3. No, the intent and actions expressed in text are an example of language understanding and not named entity recognition. Language understanding extracts the overall ­meaning from the text.
4. Analyzing social media for a brand is an example of where you would use sentiment analysis to determine the positive and negative mentions of a brand.
5. Transcription of a recording into text is an example of speech recognition. Speech ­recognition can convert audio into text.
6. Mining customer opinions can be performed by Text Analytics using the sentiment analysis operation. Sentiment analysis explores customers’ perceptions of products or services.
7. You need to add sample utterances to train the LUIS model.
8. To use a Language Understanding (LUIS) model to find the intent of a text statement, you need the ID of the LUIS app and the endpoint and key for the prediction resource.
9. The Speech to Text API can be used synchronously (real-time) or asynchronously (batch). Batch Speech to Text can translate large volumes of speech audio recordings stored in Azure Blob Storage.
10. You can choose from neutral voices, standard voices, or a custom voice. The Text to Speech API can create custom voices.