

Preparing for AI-102 - Summary of all 6 videos

Links

Preparing for AI-102 - Plan and manage an Azure AI solution (Part 1 of 6) <https://learn.microsoft.com/en-us/shows/exam-readiness-zone/preparing-for-ai-102-plan-and-manage-an-azure-ai-solution> Preparing for AI-102 - Implement decision-support solutions (Part 2 of 6) <https://learn.microsoft.com/en-us/shows/exam-readiness-zone/preparing-for-ai-102-implement-decision-support-solutions> Preparing for AI-102 - Implement Azure AI vision solutions (Part 3 of 6) <https://learn.microsoft.com/en-us/shows/exam-readiness-zone/preparing-for-ai-102-implement-azure-ai-vision-solutions> Preparing for AI-102 - Implement natural language processing solutions (Part 4 of 6) <https://learn.microsoft.com/en-us/shows/exam-readiness-zone/preparing-for-ai-102-implement-natural-language-processing-solutions> Preparing for AI-102 - Implement knowledge mining and document intelligence solutions (Part 5 of 6) <https://learn.microsoft.com/en-us/shows/exam-readiness-zone/preparing-for-ai-102-implement-knowledge-mining-and-document-intelligence-solutions> Preparing for AI-102 - Implement generative AI solutions (Part 6 of 6) <https://learn.microsoft.com/en-us/shows/exam-readiness-zone/preparing-for-ai-102-implement-generative-ai-solutions>

Tips

You should expect more questions about the "Implement natural language processing solutions" because it is 30-35% of the exam.

Plan and manage an Azure AI solution

1. Select the appropriate Azure AI service
2. Plan, create, and deploy an Azure AI service
3. Manage, monitor, and secure an Azure AI service

Select the appropriate service for an Azure AI vision solution

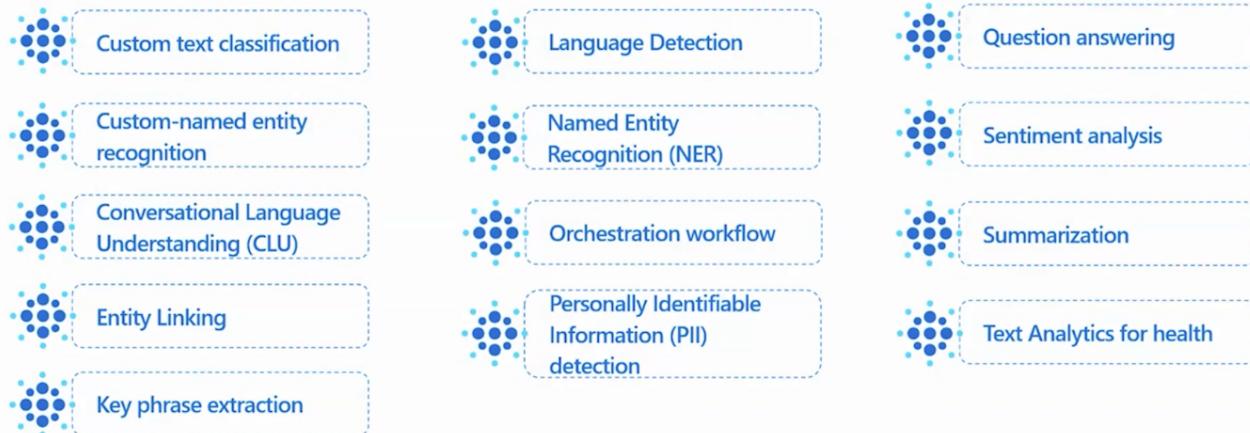
Key services that revolve around imaging or visual content

The diagram illustrates five key Azure AI vision services:

- Azure AI Vision**: Processes images and returns information.
- Face API**: Detects and recognizes faces.
- Azure AI Custom Vision Service**: Builds custom image classifiers.
- Document Intelligence**: Extracts text, key-value pairs, and tables from documents.
- Azure AI Video Indexer**: Extracts insights from video.

©Microsoft Corporation. All rights reserved.

Select the appropriate service for a natural language processing solution



Select the appropriate Azure AI services for a speech solution

The speech solution is designed to ensure apps can recognize and decipher written and spoken text.



Speech-to-text	Text-to-speech	Speech translation
Supports intermediate results, end-of-speech detection, automatic text formatting, and profanity masking, and includes real-time speech-to-text and batch transcription	Converts text to natural-sounding speech	Translates streaming audio in real-time and provides result as text/synthesized speech
Language identification Identifies languages spoken in audio when compared against supported languages	Speaker recognition Identifies and verifies the people speaking based on audio	Intent recognition Derives user intents from transcribed speech and act on voice commands
	Pronunciation assessment Evaluates pronunciation and provides feedback on the accuracy and fluency of speech	

Select the appropriate service for a document intelligence solution

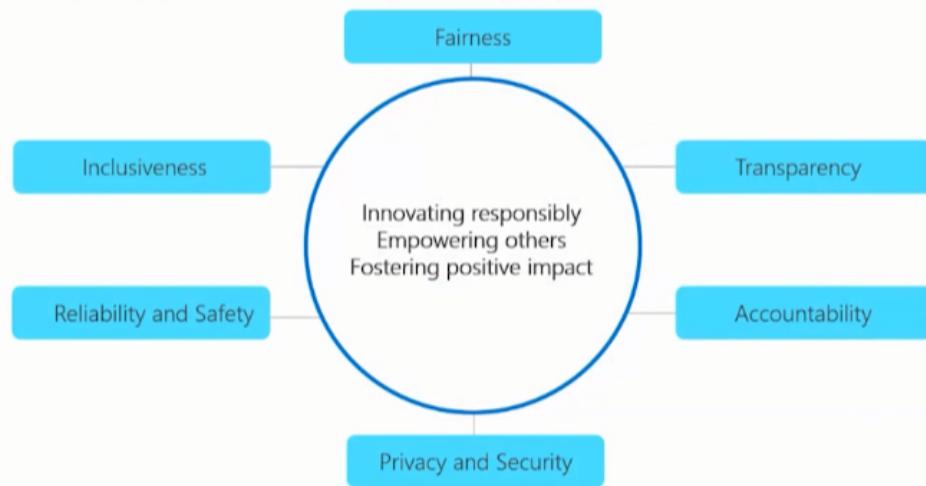
The AI Document Intelligence Studio helps in extracting texts, key-value pairs, tables, and structures automatically from documents.

The three types of models are:



Plan for a solution that meets Responsible AI principles

Plan for a solution that puts people first and meets responsible AI principles.



The following is crucial:

Create an Azure AI resource

How to create an Azure AI resource:

-  [Using the Azure portal](#)
-  [Using Azure CLI](#)
-  [Using the client library](#)
-  [Using an ARM template/Bicep/Terraform](#)

You can access Azure AI through two different resources:



Multi-service resource:

- Access multiple Azure AI resources with a single key and endpoint.
- Consolidate billing from the services you use.



Single-service resource:

- Access a single Azure AI resource with a unique key and endpoint for each service created.
- Use the free tier to try out the service.

©Microsoft Corporation. All rights reserved.

1.2

Protect account keys by using Azure Key Vault and manage authentication for an Azure AI service resource

Protecting account keys

Before you add your credential information to your Azure key vault, you need to retrieve it from your Azure AI services resource.

Key Vault reduces the chances that secrets may be accidentally leaked because you won't store security information in your application.

There are three ways to authenticate a request, each with different requirements.

Authenticate with...

- **Single-service or multi-service subscription key**
Authenticate requests with subscription keys for a specific service (for example, Azure AI Translator)
 - Single-service keys are tied to a specific service.
 - A multi-service key can be used to authenticate requests for multiple Azure AI.
- **A token**
Text Translation API and the Speech Services (Speech-to-text REST API and Text-to-speech REST API) require authentication tokens.
- **Azure Active Directory (AAD)**
Use this in more complex scenarios that require Azure role-based access control (RBAC).

Plan and implement a container deployment

Decide to deploy as a standalone Docker container or within a Kubernetes environment.



Azure Container Instances

Why deploy here?

Run containers on-demand with minimal setup in a serverless environment



Azure Kubernetes Services

Why deploy here?

If the application has multiple moving parts/components, Kubernetes enables:

- Scripted development
- Easy container scaling

Containers provide an immutable infrastructure for application packaging and deployment.

Configuring diagnostic logging

You can choose to send platform logs and metrics to a logging destination in your Azure AI resource.

**Diagnostic information
can be sent to...**



Log Analytics Workspace

- Analyzes logs and metrics with other monitoring data collected by Azure Monitor
- Leverages other Azure Monitor features

Event Hubs

- Enables user to stream data to external systems (for example, 3rd party SIEMs, other log analytics)

Azure Storage

- Archives logs and metrics to an Azure storage account for an indefinite amount of time
- Less expensive

©Microsoft Corporation Azure

1.3

Monitor an Azure AI resource

The Monitoring section in your Azure AI resource includes four areas that you can use for monitoring your resource.



Metrics

Capture regular metrics in near real time and store these in a time-series database



Alerts

Configure notifications and alerts for key systems and applications



Diagnostic settings

Configure diagnostics settings to provide detailed information for diagnostics and auditing



Activity Logs

Contains time-stamped information about changes made to resources

©Microsoft Corporation Azure

1.3

Implement decision-support solutions

1. Create decision-support solutions for data monitoring and content delivery

At most 10.000 intervals forward counting from the most recent timestamp

Implement a data monitoring solution with Azure AI Metrics Advisor

Azure AI Metrics Advisor uses AI for monitoring data and detecting anomalies in time series data.



Implement a text moderation solution with Azure AI Content Safety

1

Analyze Text Content

- 1 Replace <endpoint> with the URL to your resource's endpoint.
- 2 Replace <your_subscription_key> with the key in your resource.
- 3 Replace the "text" field in the body with the text to be analyzed.

2

Interpret the API response

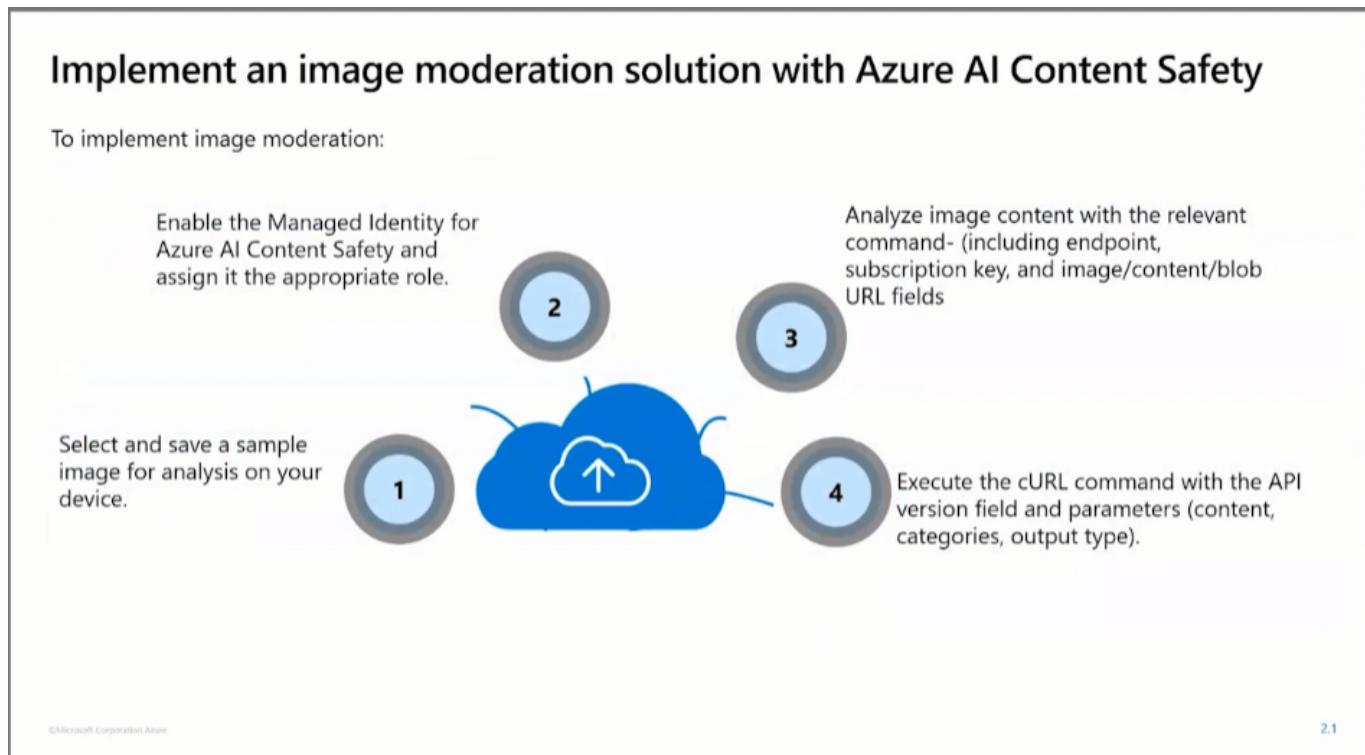
The text moderation results should be displayed as JSON data in the console output.

Harm Categories

- Hate and fairness
- Sexual
- Violence
- Self-harm

Severity Level

- Level 0 (Safe)
- Level 4 (Medium)
- Level 1
- Level 5
- Level 2
- Level 6 (High)
- Level 3 (Low)
- Level 7

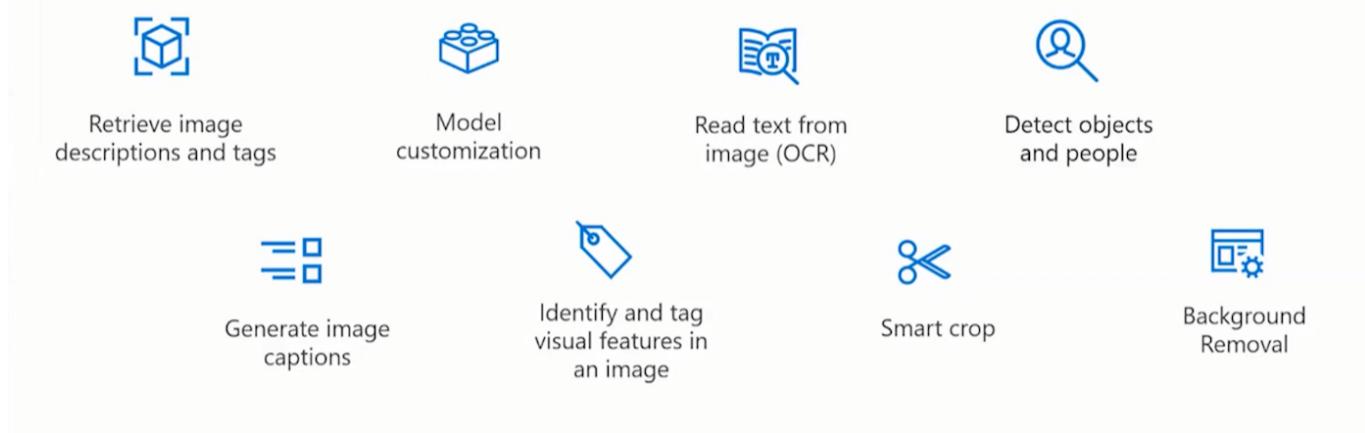


Implement Azure AI vision solutions

1. Analyze images
2. Implement custom Azure AI vision models by using Azure AI vision
3. Analyze videos

Select visual features to meet image processing requirements

The Azure AI Vision Image Analysis service can be used to extract several features from images.



For a custom vision model, we will need to create a new project and choose between classification and model detection. We can then choose tags, domains, etc...

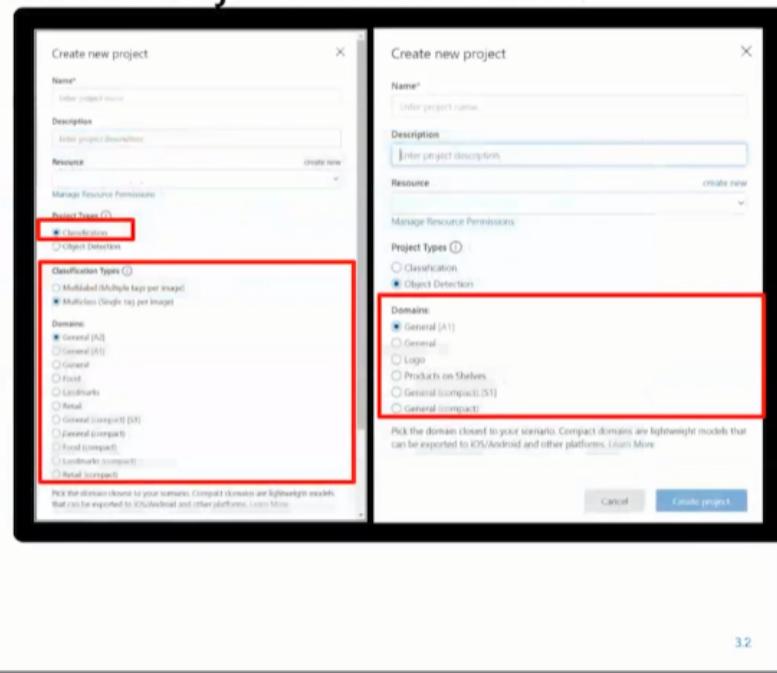
Choose between image classification and object detection models

You can use Azure AI Custom Vision to create either an:

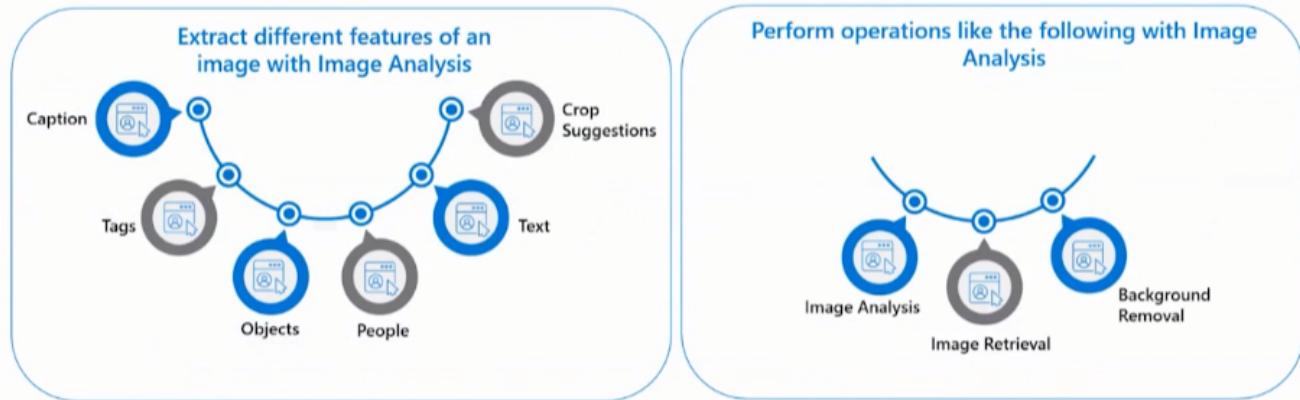
- Image classification model to classify images per classes or labels
- Object detection model to detect objects in an image

To do so:

- 1 Create a new project by selecting New Project.
- 2 Enter a name and a description (optional) for the project.
- 3 Under Project Type, choose either Classification or Object Detection.
 - Note: For Classification, you will need to further select the Classification Type as Multilabel to add more than one tag to images or Multiclass, to add only one.
- 4 Select an available domain and then select Create.



Include image analysis features in an image processing request and interpret image processing responses



We should know the difference between image analysis, image retrieval and background removal.

For OCR, we can use two methods:

Extract text from images and convert handwritten texts using AI Azure Vision

Methods to extract text from images:

	Extract text from images	Azure AI Vision provides a synchronous API which aids in embedding OCR in user experience and optimize text extraction from general, non-document images.
	Extract text from images	Azure AI Vision provides a synchronous API which aids in embedding OCR in user experience and optimize text extraction from general, non-document images.

Steps to extract text:

Step 1	Set up the application, create environment variables, and find the key and endpoint.
Step 2	From the project directory, open the relevant file and paste the code required to analyze it.
Step 3	Run the application and view the output.

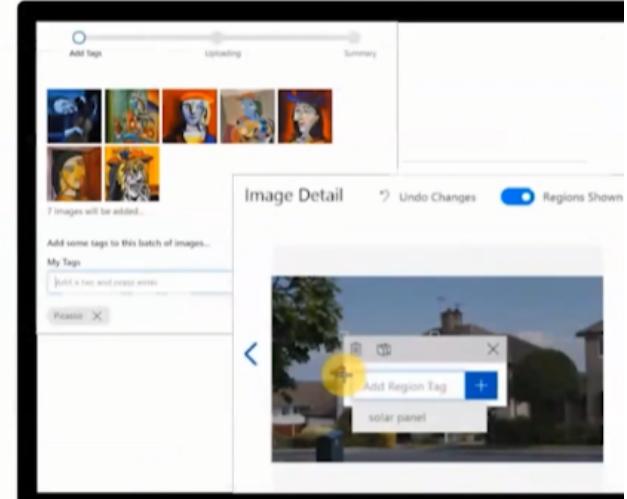
About labeling, remember that we need to use bounding boxes. Some tips are to use clear and consistent tags, use high quality and varied images, and review the labels.

Label images

You can upload and tag your images manually to train the classifier or detector.

Steps to label images

- ① Under **Training Images**, select the + sign.
- ② Choose the images to upload and select **Upload**.
- ③ Select the **Untagged** section to show your untagged images.
 - a. For image classification, enter text into the **My Tags** field, press **Enter**, and finish uploading the images with the **Upload [number]** files and **Done** buttons.
 - b. For image detection, click and drag a rectangle around the object in your image and enter a tag (or choose an existing one) and move to the next image.

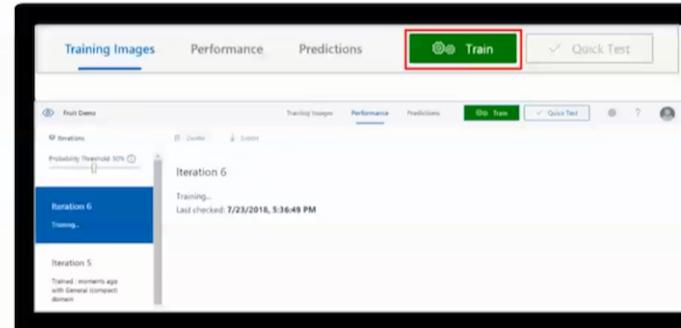


About training:

Train a custom image model, including image classification and object detection

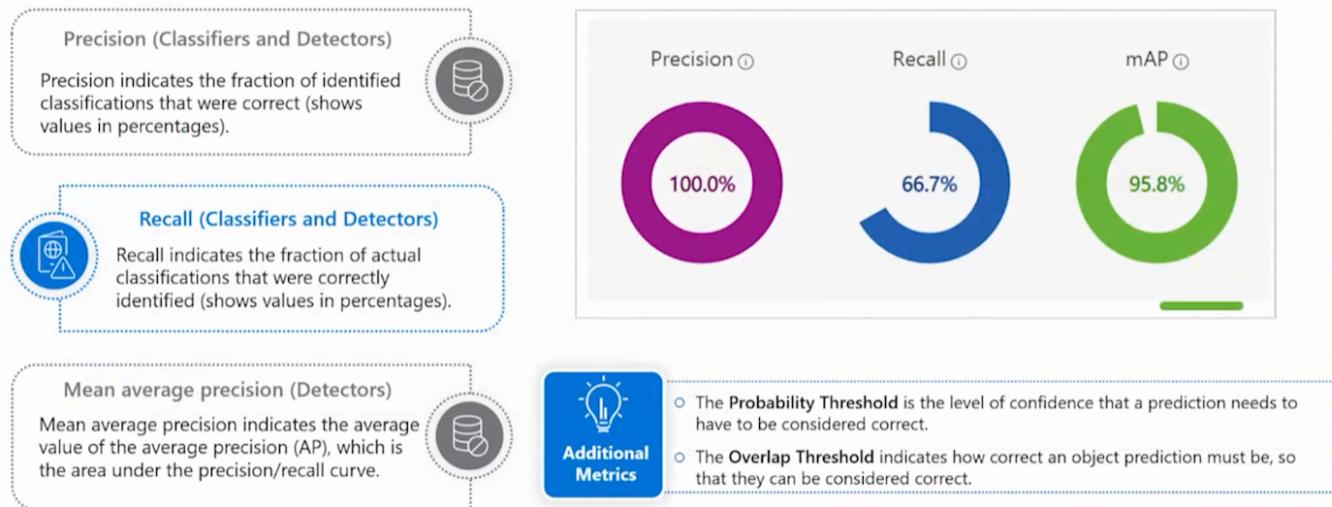
Training the classifier/ detector

- 1 Select the **Train** button to train the classifier or detector.
- 2 View information about the training process, such as number of iterations in the **Performance** tab.
- 3 Use the **Delete** button to delete an iteration if it's obsolete.



And evaluation:

Evaluate Azure AI custom vision model metrics



The keyword for recall is actual classification. mAP is mean Average Precision!

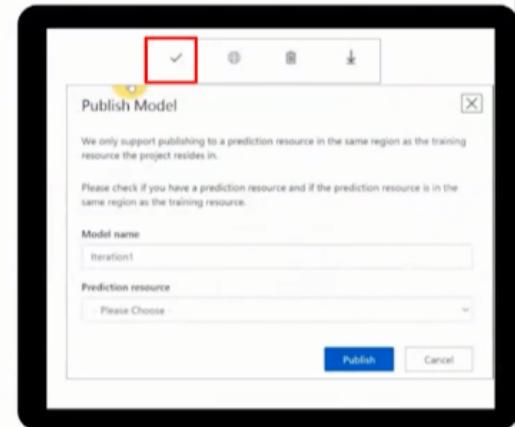
Publish and consume an Azure AI custom vision model

You can deploy a model, so it is accessible from the API.

Step 1	Select the Publish (✓) button from the top of the Performance tab.
Step 2	In the Publish Model pop-up window, specify the Model name and Prediction resource .
Step 3	Select Publish .

Consuming an Azure AI custom vision model:

- Prevent overfitting
- Increase the quantity of data
- Balance labeling of images
- Use a variety of data
- Use negative samples for image classifiers
- Include truncated images and occluded objects for image detectors
- Use prediction images for further training
- Visually inspect predictions



©Microsoft Corporation. All rights reserved.

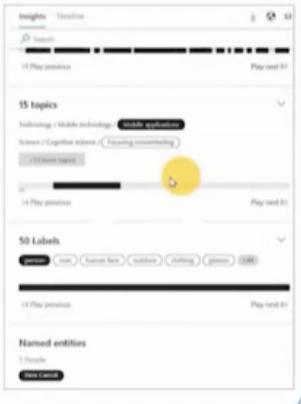
3.2

It can help us identify spoken words, written text, faces, topics, etc...

Use Azure AI Video Indexer to extract insights from a video or live stream

Upload a video in Azure AI Video Indexer to view the following aspects of the video:

Insights



On the **Insights** pane, you can view different insights from the video such as the **Topics**, **Labels**, **Named entities**, **Emotions** and **Scenes**.

Timeline

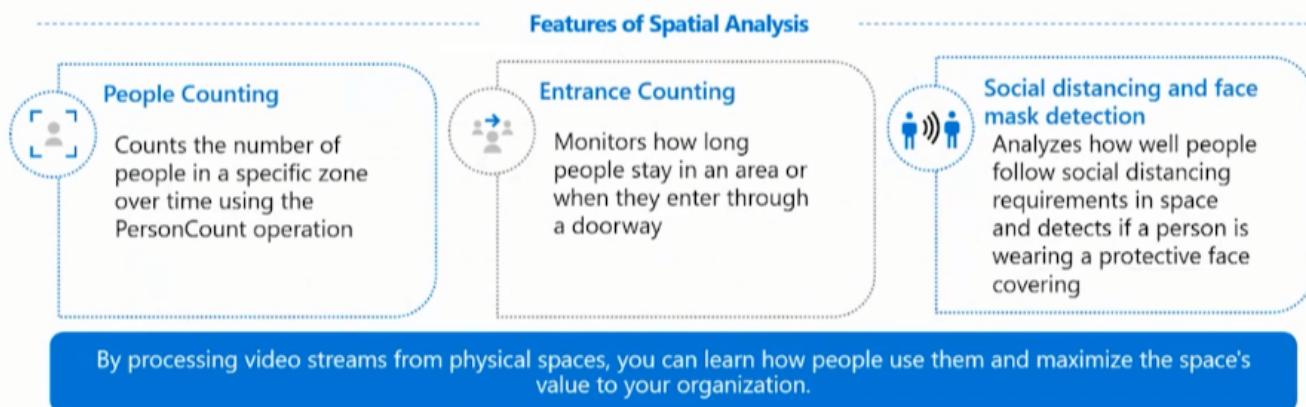


On the **Timeline** pane, you can view the time stamps from the video along with the dialogue in these instances.

©Microsoft Corporation. All rights reserved.

2.4

Use Azure AI Vision Spatial Analysis to detect presence and movements of people in video



Implement natural language processing solutions

1. Analyze text by using Azure AI Language
2. Process speech by using Azure AI Speech
3. Translate language
4. Implement and manage a language understanding model by using Azure AI language
5. Create a question-answering solution by using Azure AI language

Extract key phrases

Key phrase extraction is used to quickly identify the main concepts in text.

Steps to extract key phrases:

Step 1

Create an Azure AI Language resource, which will generate a password or key and an endpoint URL for API request authentication.

Step 2

Create a request using either the REST API or the client library.

Step 3

Send the request with the raw unstructured text. Use the key and endpoint for authentication.

Step 4

Stream or store the response locally. Analysis will be performed on the raw data as is, with no additional customization needed.

Three ways to use key phrase extraction

- With Language Studio
- With REST API or Client library
- With Docker Container

An entity is a specific object or concept that can be identified in the text. It is not the same as a key phrase.

Extract entities

Entity Linking: the ability to identify and disambiguate the identity of an entity found in text

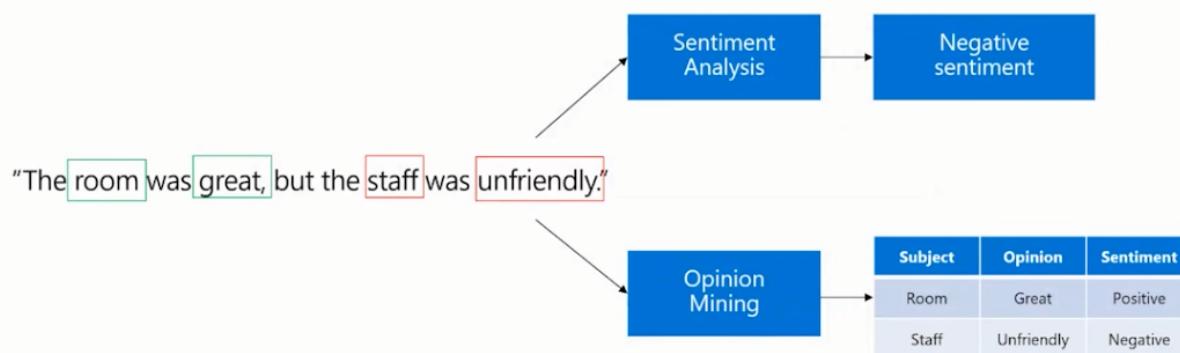


Named Entity Recognition (NER): the ability to identify different entities in text and categorize them into pre-defined classes or types such as: person, location, event, product and organization

We have separated endpoints for entity linking, and you can get information from multiple sources (like Wikipedia)

Determine sentiment of text

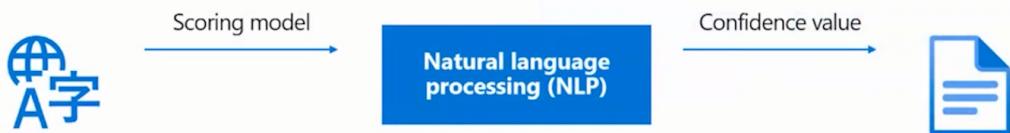
The Azure AI Language's Sentiment Analysis feature evaluates text and returns sentiment scores and labels for each sentence.



This gives us the CODE of the language (if it is spanish, english, etc...). It can detect the largest language represented if there are multiple in the document.

Detect the language used in text

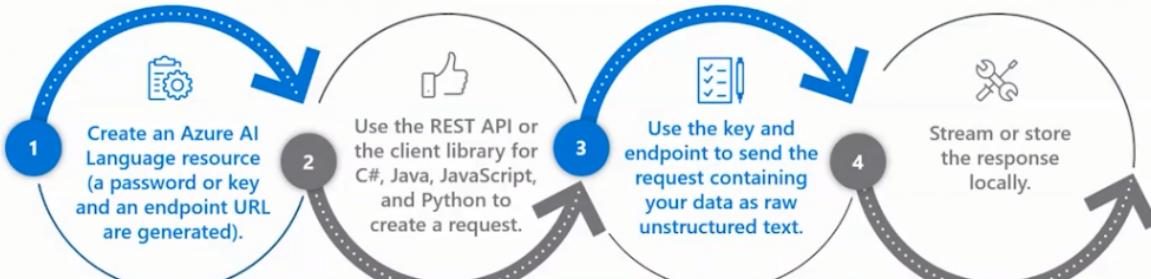
The Language Detection feature evaluates text input and returns language identifiers with a score that indicates analysis strength.



Input/output cases	Mixed-language content Returns the language with the largest representation but a low confidence	Ambiguous content Returns a countryHint response based on limited context
--------------------	---	--

Detect personally identifiable information (PII) in text

Personally Identifiable Information (PII) detection can be used to identify, categorize, and redact sensitive information in unstructured text.



You should:

- Use Language Studio to enable entity linking with text examples without the need for an Azure account and permit use of your own data when you sign up.
- Use the REST API or client library to integrate PII detection into your applications.

About text-to-speech

Implement text-to-speech

The Speech Services speech synthesis process generates artificial reproduction of human speech and converts language text into speech.



Implement speech-to-text and improve text-to-speech by using Speech Synthesis Markup Language (SSML)

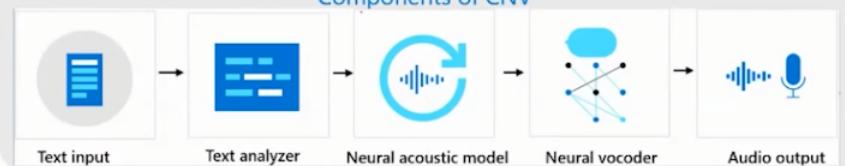
Speech-to-text: Provides real-time transcription of audio streams based on Machine Learning (ML) and AI.



SSML can be used to fine-tune the text-to-speech output attributes (pitch, pronunciation, speaking rate, and volume).

Custom Neural Voice (CNV) creates one-of-a-kind, customized, synthetic voices

Components of CNV

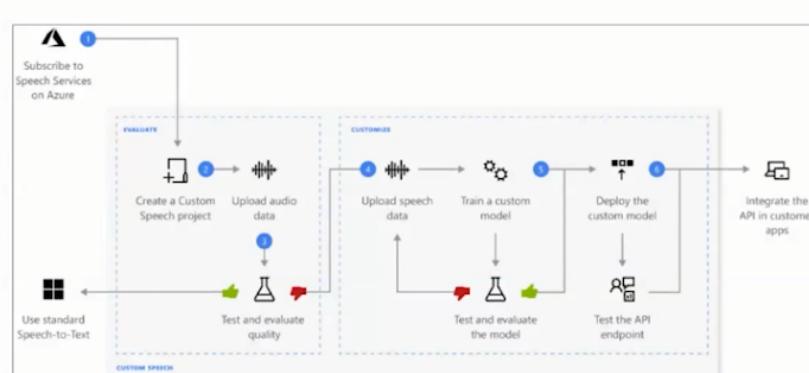


With CNV, build highly natural-sounding voices by providing human speech samples as training data.

SSML is based on XML.

Implement custom speech solutions

Test a baseline or custom language model for Word Error Rate (WER) using accuracy tests or a custom acoustic model.



Reduce WER score each iteration

$$WER = \frac{I + D + S}{N} * 100\%$$

Ready to use	5-10%
Acceptable	20%
Needs additional testing	30%+

Implement intent recognition

Intent recognition can be used to determine what the user wants to initiate or do.

Use Pattern matching for a quick offline solution.

Steps

- 1 Create code and speech configuration.
- 2 Initialize an intent recognizer and declare entities as intents.
- 3 Enable recognition of intent.
- 4 Instruct code to stop upon intent recognition.
- 5 Display recognition results.
- 6 Build and run the application.

Use CLU to build a custom natural language understanding model to predict the intent of incoming instances.

Steps

- 1 Create a project by importing a JSON file.
- 2 Train your model.
- 3 Choose your training mode and data splitting method and select Train.
- 4 Deploy your model.
- 5 Use the model to recognize intents from a microphone.

Implement keyword recognition

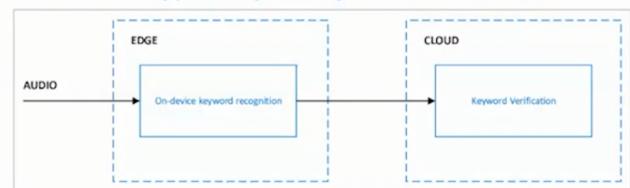
Keyword recognition can be used to detect a word or short phrase within any audio.

To implement keyword recognition



- 1 Create a new project in Speech Studio and enter the project details.
- 2 To create a custom keyword, select **Create a new model** and enter the **Name**, **Description**, and **Keyword** and select **Next**.
- 3 Listen to the candidate pronunciations generated by the portal and remove incorrect ones, if any. Select **Next**.
- 4 Select a model type and select **Create**.
- 5 Select **Tune** from the collapsible menu to download the model.
- 6 This model can now be used on any file to detect the keyword.

A typical keyword system consists of:



Keyword models can be:

- **Basic:** They may not have optimal accuracy characteristics and are suited for rapid prototyping purposes.
- **Advanced:** They improve accuracy characteristics and are suited for product integration purposes.

Translate text and documents by using the Azure AI Translator service

Text translation can be used for quick and accurate source-to-target text translation in real time across all supported languages.

Text Translation enables the following operations

- Languages
- Translate
- Transliterate
- Detect
- Dictionary lookup
- Dictionary example

Document Translation enables the following operations

- Translate large and numerous files
- Perseverance
- Custom translation/glossaries
- Detect
- Translate multiple languages

Text Translation Deployment

- To add text translation to your projects, use the:
- REST API cloud-based Translator in Azure
 - REST API translate request with the Text translation Docker container

Document Translation Deployment

- To add document translation to your projects, use the:
- REST API cloud-based Translator in Azure
 - Client-library SDK to asynchronously translate whole documents in supported languages and formats

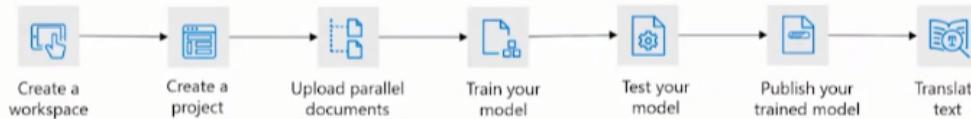
Import to differentiate between Text Translation and Document Translation.

Transliterate -> English to Japanese (hiragana/katakana). Not the same as translation.

Implement custom translation, including training, improving, and publishing a custom model

Use custom translation to get better translations, be productive and cost effective, and securely translate anytime, anywhere, on all apps/services.

Custom Translation process flow



To train your model:

1. Select Train model, enter the sample data, and select full training.
2. Select sample-source language-target language and review the training cost.
3. Select Train now, then Train to confirm.
4. Once the model is trained, select Model details to review details.

To test and publish your model:

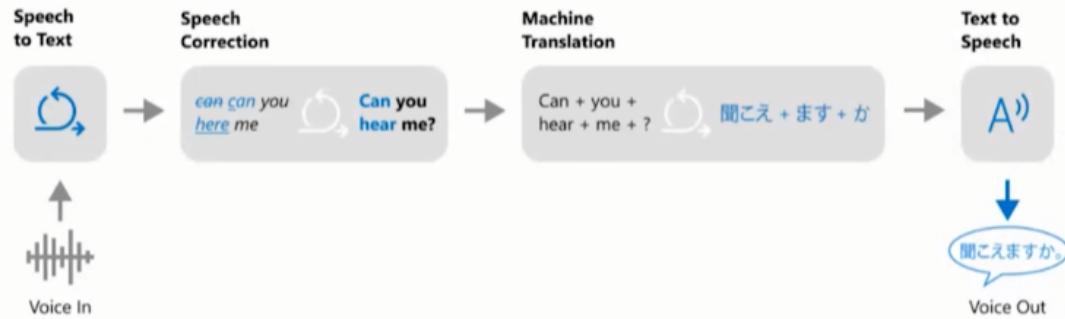
1. Select Test model and enter sample data.
2. Test (human evaluate) the translation.
3. Select Publish model, enter sample data and select Publish.
4. Check the desired region(s) and select Publish.

©Microsoft Corporation Azure

4.3

Translate speech-to-speech by using the Azure AI Speech service

The speech-to-speech service can accept audio input and translate that audio to a different language.



©Microsoft Corporation Azure

4.3

Translate speech-to-text and multiple languages simultaneously by using the Azure AI Speech service

There are four translation services available for translating speech-to-text



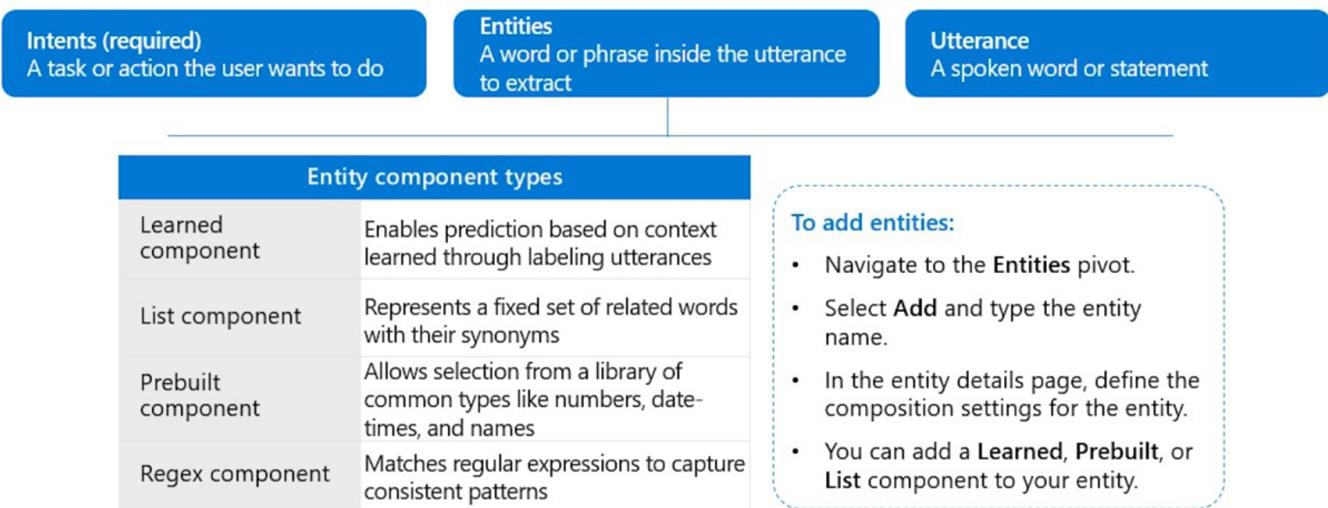
Translating documents from and to multiple languages

- Use the Document Translation feature of Azure AI Translator to translate entire documents, or batches of documents, in various file formats.
- Use it to **translate documents into multiple languages** with just a single request.
- Use the Autodetect feature to **translate documents with content in multiple languages into your target language**. (The Azure blob storage container stores the documents within a specific folder.)

This one is important:

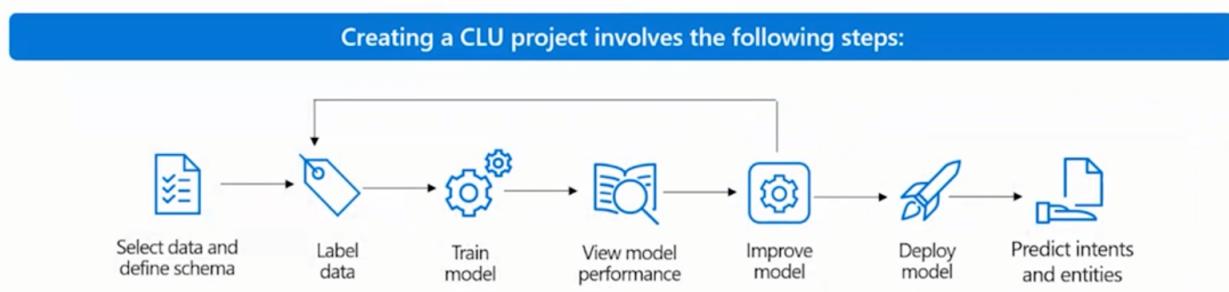
Create intents, entities and add utterances

Conversational Language Understanding (CLU) makes use of three key aspects for understanding language.

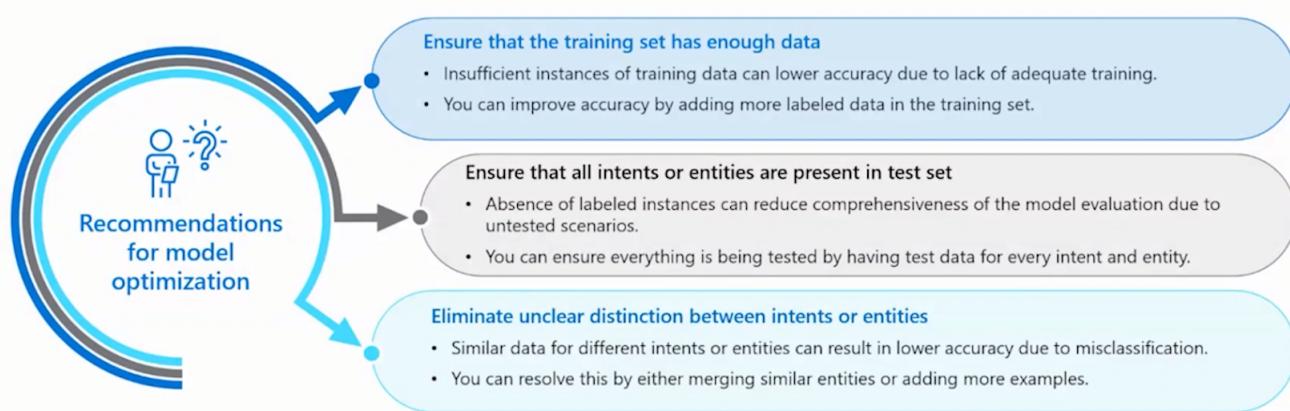


Train, evaluate, deploy, and test a language understanding model

Conversational language understanding (CLU) can be used to build custom natural language understanding models which predict the intention of an incoming utterance and extract any important information from it.

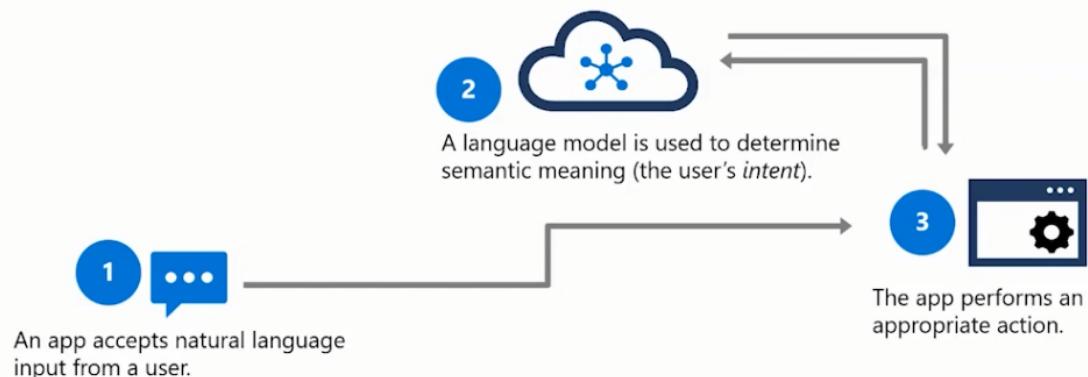


Optimize a Language Understanding model

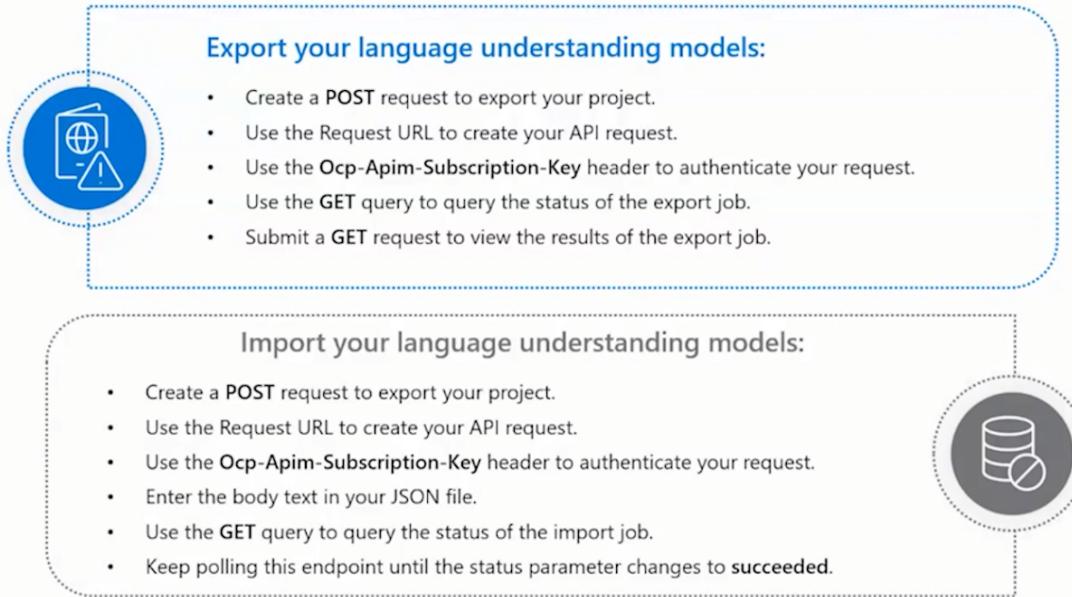


Consume a language model from a client application

Azure AI-Language provides Natural Language Processing (NLP) features for understanding and analyzing text.



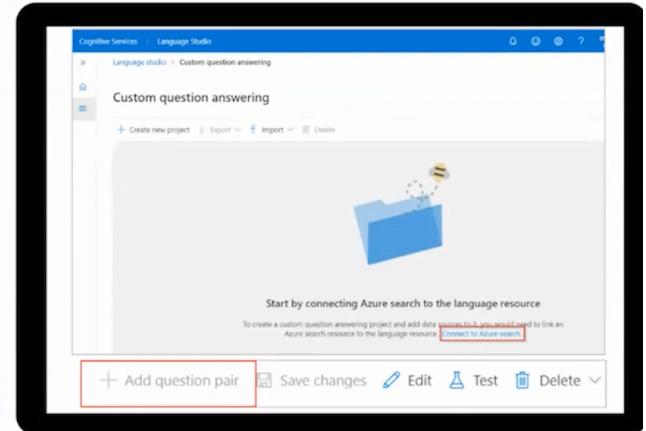
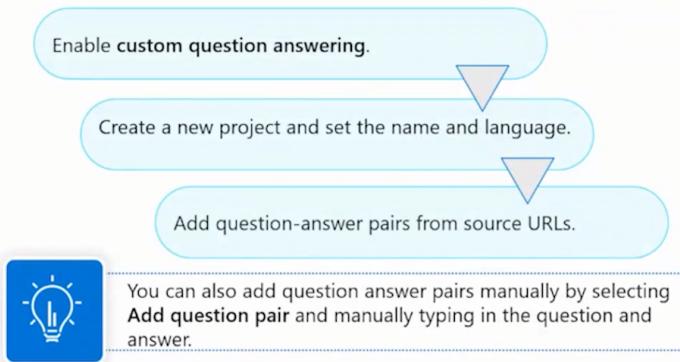
Import and export language understanding models



Create a question-answering project and add question-and-answer pairs manually

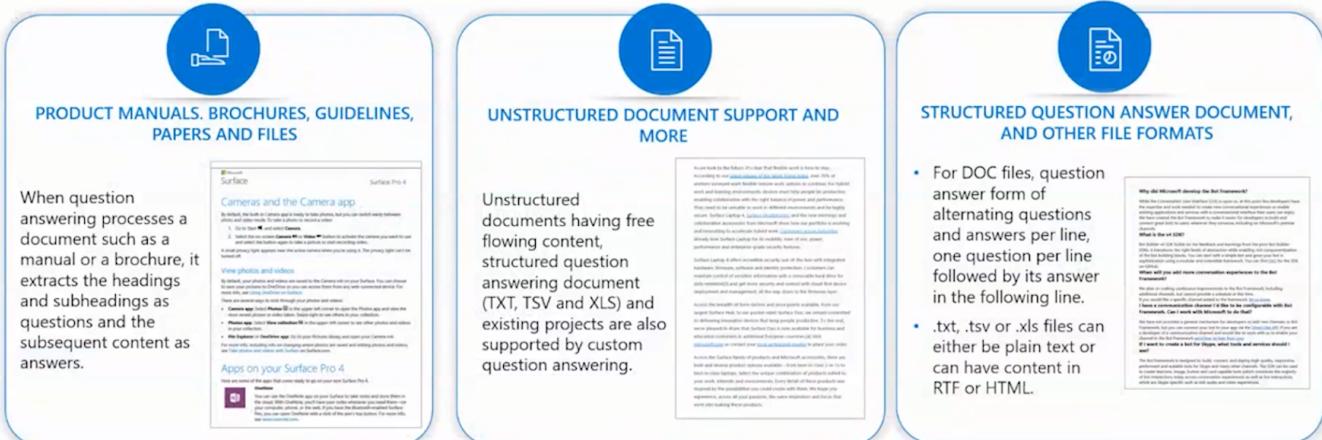
Question answering can be used to build conversational client applications, such as social media applications, chat bots, and speech-enabled desktop applications.

Stages in creating a new question answering project



Import sources

You can use Azure Question Answering to add question and answer pairs from different documents:

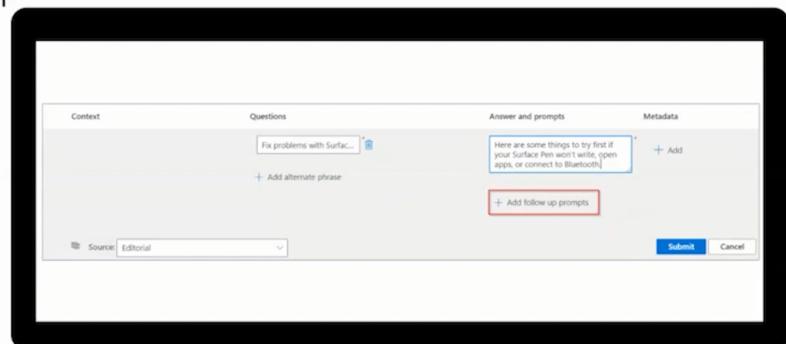


Create a multi-turn conversation

Question answering provides multi-turn prompts, which can connect question and answer pairs.

You can add a follow-up prompt to a newly created question pair by following these steps:

- 1 Select Add follow-up prompts.
- 2 Fill in the details of the prompt.
- 3 Select Create link to new pair and then select Done.
- 4 Select Save changes.
- 5 You can add multiple prompts to the same question by repeating the process.



Add chit-chat to a knowledge base

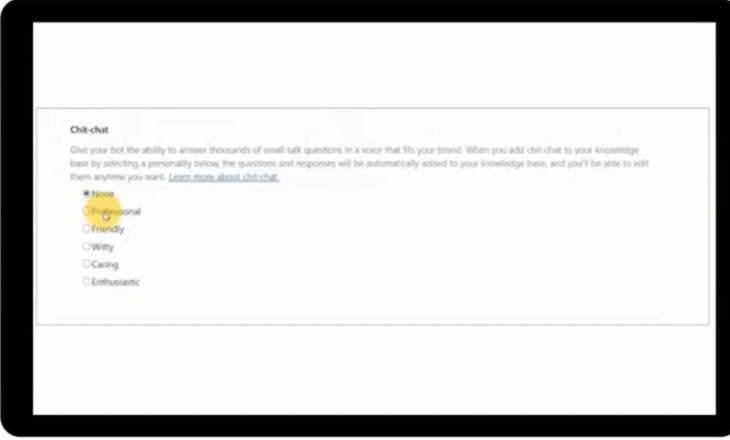
You can use chit-chat to make your bot conversational and engaging by selecting a personality for your conversation.



Give your bot the ability to answer questions in a way that fits your brand.



Set a personality for your conversation and have question and answers automatically added to your knowledge base.



Export a knowledge base

You may want to export a knowledge base for several reasons:

- To implement a backup and restore process
- To integrate with your CI/CD pipeline
- To move your data to different regions

Step 1

From the Answer questions section, select Open custom question answering.

Step 2

Select the project you want to export and then select Export.

Step 3

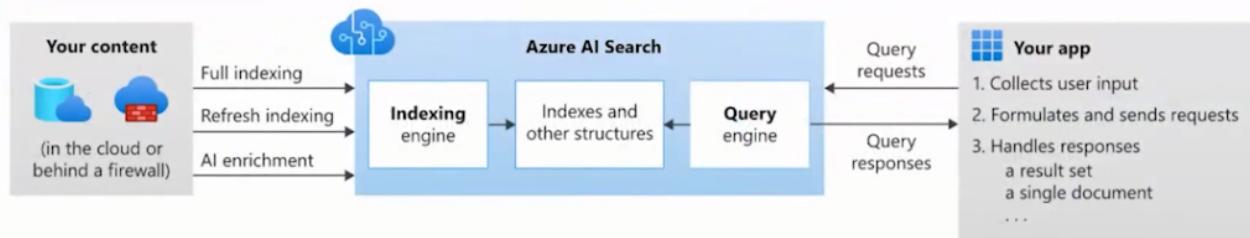
Select if you want to export the file as an Excel file or a TSV file. The file is saved locally as a zip file.

Implement knowledge mining and document intelligence solutions

1. Implement an Azure AI Search solution
2. Implement an Azure AI Document Intelligence solution

Provision a Cognitive Search resource

Build a rich search experience over private, heterogeneous content in web, mobile, and enterprise applications.

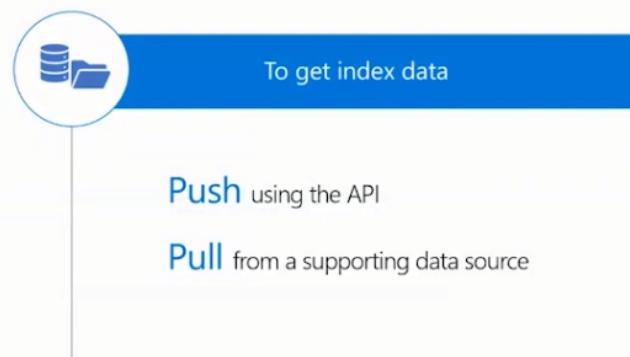


Create data sources

To pull the data from a store and populate the index, you must define a data source in your Azure AI Search resource.

Data sources can reference:

- Azure Storage (blob or table)
- Azure Cosmos DB
- Azure SQL Database, SQL Managed Instance, or SQL server in a VM



This one is important:

Create an index

To provide information through search, you must define an index that contains the fields used to query, filter, and sort data.

An index consists of a collection of JSON objects, each with one or more fields including a unique key.



Different index attributes

- | | |
|---|--|
| <input checked="" type="radio"/> Searchable | <input checked="" type="radio"/> Facetable |
| <input checked="" type="radio"/> Filterable | <input checked="" type="radio"/> Retrievable |
| <input checked="" type="radio"/> Sortable | |

Potential field types: text, numbers, DateTime values, lists, and complex structures

Define a skillset

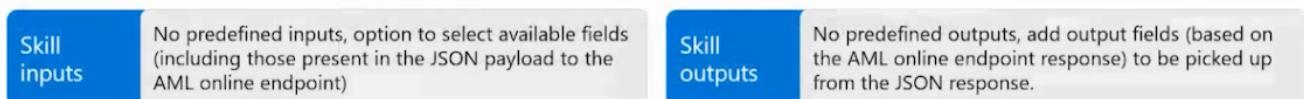
Create a skillset with the following key steps:



Implement custom skills and include them in a skillset

Use these parameters to implement custom skills in a skillset:

Parameter name	Description
uri	Required for key authentication
key	Required for key authentication
resourceId	Required for token authentication
region	Optional for token authentication
timeout	When specified, indicates the timeout for the http client making the API call
degreeOfParallelism	When specified, indicates the number of calls the indexer makes in parallel to the endpoint provided



Create and run an indexer

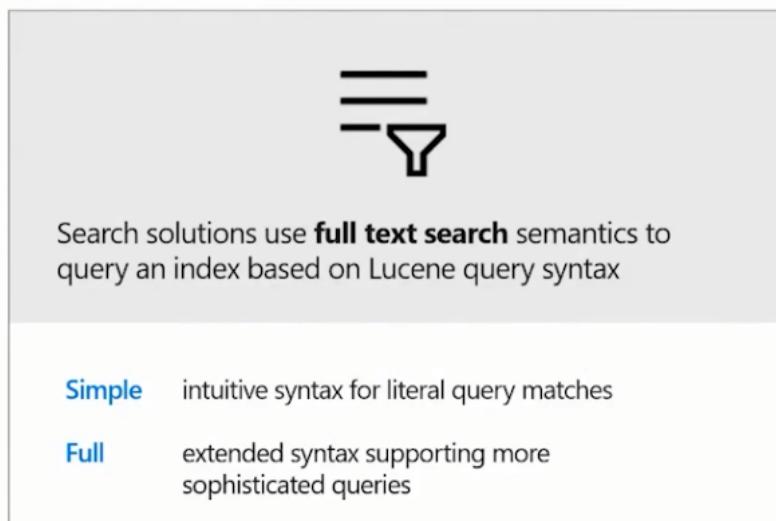
Indexers use a predefined *data source* and *index* to establish an indexing pipeline that extracts and serializes source data, passing it to a search service for data ingestion.

Request body

```
JSON
{
  "name" : (optional on PUT; required on POST) "Name of the indexer",
  "description" : (optional) "Anything you want, or nothing at all",
  "dataSourceName" : (required) "Name of an existing data source",
  "targetIndexName" : (required) "Name of an existing index",
  "skillsetName" : (required for AI enrichment) "Name of an existing skillset",
  "schedule" : (optional but runs once immediately if unspecified) { ... },
  "parameters" : (optional) { ... },
  "fieldMappings" : (optional) { ... },
  "outputFieldMappings" : (required for AI enrichment) { ... },
  "encryptionKey":(optional) { },
  "disabled" : (optional) Boolean value indicating whether the indexer is disabled. False by default.
}
```

Query an index, including syntax, sorting, filtering, and wildcards

After you have created an index, you can query it to search for information in the indexed document content.



Queries are processed in four stages:

- 1 Query parsing
- 2 Lexical analysis
- 3 Document retrieval
- 4 Scoring

Manage Knowledge Store projections, including file, object, and table projections

A projection lets you “project” your data into a shape that aligns with your needs.

What are projections?

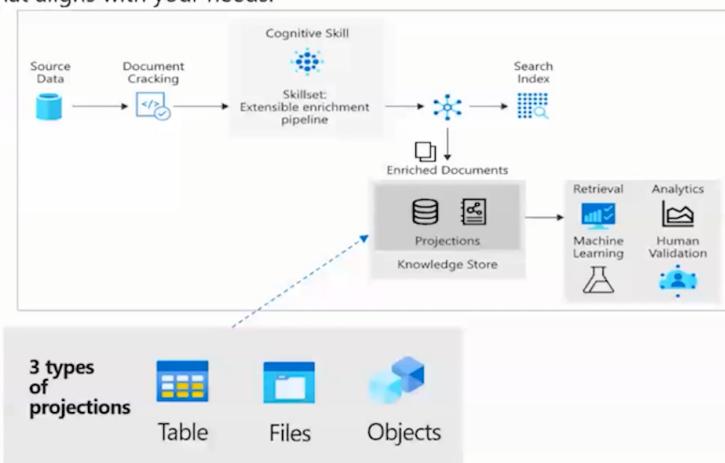
Views of enriched documents that can be saved to physical storage

Why use projections?

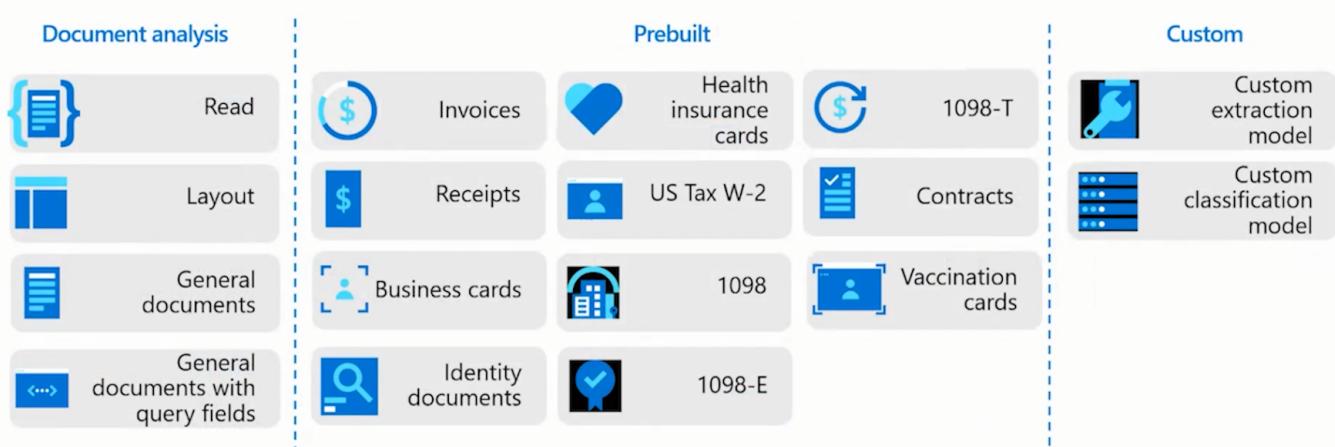
For knowledge mining purposes, so that tools can read data with no additional effort

How to use projections?

Read projected data in containers or tables specified through projections

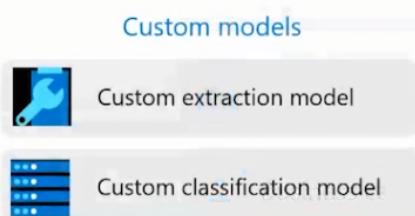


Provision a Document Intelligence resource



Implement a custom document intelligence model and train, test and publish a custom document intelligence model

Train custom models to classify documents and extract text, structure, and fields from your forms or documents.



Steps to train, test, and publish a custom document intelligence model

- 1 Create a project in the Document Intelligence Studio
- 2 Label your data
- 3 Train your model
- 4 Test the model

Implement generative AI solutions

1. Use Azure OpenAI Service to generate content
2. Optimize generative AI

Provision an Azure OpenAI Service resource

----- Deploy a model in Azure OpenAI Studio -----

- 1 Apply for Azure OpenAI service access:
<https://aka.ms/oaiapply>
- 2 Create an Azure OpenAI resource in the Azure portal.
 - i. Identify resource
 - ii. Configure network security
 - iii. Confirm configuration and create resource

You can also use Azure CLI:

```
az cognitiveservices account create \
-n MyOpenAIResource \
-g MyResourceGroup \
-l eastus \
--kind OpenAI \
--sku s0 \
--subscription subscriptionID
```

A screenshot of the 'Create Azure OpenAI' wizard. It shows the 'Basics' tab selected. The page title is 'Create Azure OpenAI ...'. Below it, there's a brief description: 'Enable new business solutions with OpenAI's language generation capabilities powered by GPT-3 models. These models have been pretrained with trillions of words and can easily adapt to your scenario with a few short examples provided at inference. Apply them to numerous scenarios, from summarization to content and code generation.' There are tabs for 'Basics', 'Network', 'Tags', and 'Review + submit'. The 'Project Details' section includes fields for 'Subscription' (dropdown), 'Resource group' (dropdown with 'Create new' option), 'Region' (dropdown set to 'East US'), 'Name' (text input), and 'Pricing tier' (dropdown).

Home > Azure AI services | Azure OpenAI >
Create Azure OpenAI ...

Basics Network Tags Review + submit

Enable new business solutions with OpenAI's language generation capabilities powered by GPT-3 models. These models have been pretrained with trillions of words and can easily adapt to your scenario with a few short examples provided at inference. Apply them to numerous scenarios, from summarization to content and code generation.

Learn more

Project Details

Subscription * Resource group * Create new

Region East US

Name *

Pricing tier *

Select and deploy an Azure OpenAI model

Model Family	Description
GPT-4	Newest, most capable chat-based models for language and code generation (<i>restricted</i>)
GPT-3.5	Natural language and code-generation models
Embeddings	Models that use embeddings for specific tasks (similarity, text search, and code search)
DALL-E	Image-generation model (<i>restricted preview</i>)

Deploy a model in Azure OpenAI Studio

- 1 Select the subscription and Open AI resource to use.
- 2 Create a new deployment:
 - i. Select a model
 - ii. Add a deployment name
 - iii. Setting advanced features such as content filters and/or tokens per min rate limit

You can also use Azure CLI:

```
az cognitiveservices account deployment create \
-g myResourceGroupName \
-n MyOpenAIResource \
--deployment-name my-gpt-model \
--model-name gpt-35-turbo \
--model-version "0301" \
--model-format OpenAI \
--scale-settings-scale-type "Standard"
```

Submit prompts to generate natural language

Task	Prompt	Completion
Classifying content	Tweet: I enjoyed the training course. Sentiment:	Positive
Generating new content	Write a poem about databases	Databases, oh databases, You keep our information safe, From the small to the large, You store our data in a place.
Transformation/Translation	English: Hello French:	Bonjour
Summarization	Scotland is [<i>long description of Scotland...</i>] Summarize the previous text	Scotland is [<i>summarized description...</i>]
Continuation	One way to grow tomatoes is to	start with seeds...
Question answering	How many moons does Earth have?	Earth has one moon.
Chat	<i>Setup, followed by messages...</i>	<i>A sequence of relevant responses</i>

Submit prompts to generate code

Use natural language prompts to generate code



Use prompt engineering to describe what you want clearly



Break down complex tasks into smaller pieces

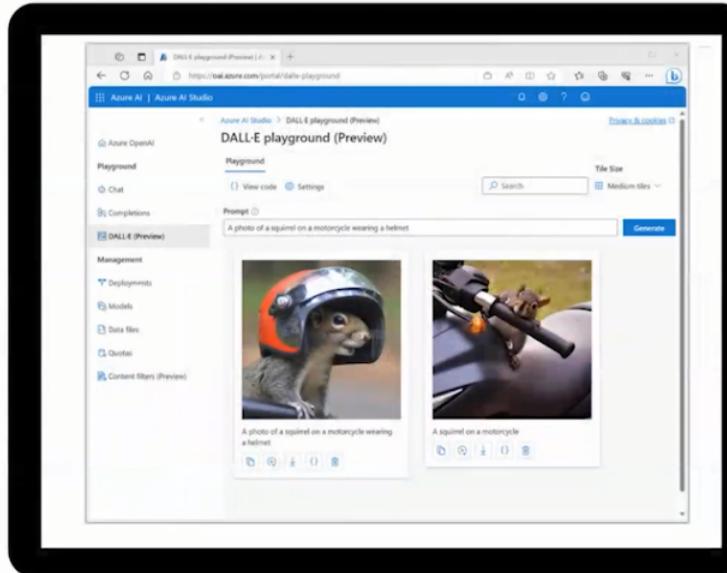


```
def binary_search(array, target):
    low = 0
    high = len(array) - 1

    while low <= high:
        mid = (low + high) // 2
        if array[mid] == target:
            return mid
        elif array[mid] < target:
            low = mid + 1
        else:
            high = mid - 1

    return -1
```

Use the DALL-E model to generate images



The screenshot shows the Azure AI Studio DALL-E playground interface. A prompt is entered: "A photo of a squirrel on a motorcycle wearing a helmet". Two generated images are displayed: one of a squirrel wearing a red and black motorcycle helmet, and another of a squirrel on a motorcycle handlebar.

Generate images with a description

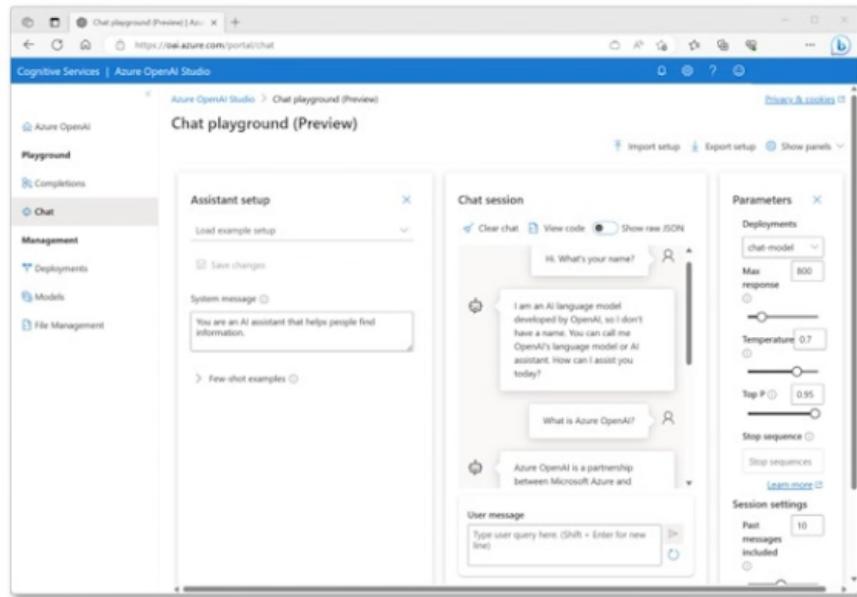
- Uses Neural network-based model for generating images
- Uses natural language to describe what the image should be
- Specifies content and style for better accuracy

Apply prompt engineering techniques to improve responses

- Providing clear instructions
- Primary, supporting, and grounding content
- Providing cues
- Requesting output composition
- Using a system message
- Conversation history and few-shot learning
- Chain of thought

Configure parameters to control generative behavior

Chat playground



Use your own data with an Azure OpenAI model



Set up your data source

- Use an existing data source
- Leverage the data already in your account (such as blob storage)



Configure the studio or your app to connect to that data source

- Point the studio to the data source and set up the connection
- Specify the data source in the prompt parameters



Use the Azure OpenAI model (with your data for grounding)

- Chat with the AI model per usual (the model will use your data source if it finds relevant information)
- Limit the AI model to use only your data source.

Fine-tune an Azure OpenAI model



Fine-tuning is a method to customize a model, like "gpt-3.5-turbo," by training it with extra data.



Fine-tuning can improve request quality beyond just using prompts, adapt the model with larger example semantic model, and reduce the need for many examples to achieve high-quality responses.



Fine-tuning is expensive and time-consuming, so it should be reserved for cases where it's essential.