

1.1 - Prepare to develop AI solutions on Azure

- [Overview](#)
 - [Introduction](#)
 - [Define artificial intelligence](#)
 - [Understand AI-related terms](#)
 - [Data science](#)
 - [Machine learning](#)
 - [Artificial intelligence](#)
 - [Understand considerations for AI Engineers](#)
 - [Model training and inferencing](#)
 - [Probability and confidence scores](#)
 - [Responsible AI and ethics](#)
 - [Understand considerations for responsible AI](#)
 - [Fairness](#)
 - [Reliability and safety](#)
 - [Privacy and security](#)
 - [Inclusiveness](#)
 - [Transparency](#)
 - [Accountability](#)
 - [Understand capabilities of Azure Machine Learning](#)
 - [Understand capabilities of Azure AI Services](#)
 - [Understand capabilities of the Azure OpenAI Service](#)
 - [Understand capabilities of Azure Cognitive Search](#)
 - [Knowledge Check](#)
 - [Summary](#)
-

Overview

As an aspiring Azure AI Engineer, you should understand core concepts and principles of AI development, and the capabilities of Azure services used in AI solutions.

Introduction

So you want to become an AI Engineer?

Your journey starts with an exploration of the core concepts and principles on which artificial intelligence is based, and an overview of the Microsoft Azure services you can use to develop AI solutions.

After completing this module, you will be able to:

- Define artificial intelligence





- Understand AI-related terms
- Understand considerations for AI Engineers
- Understand considerations for responsible AI
- Understand capabilities of Azure Machine Learning
- Understand capabilities of Azure AI Services
- Understand capabilities of Azure OpenAI Service
- Understand capabilities of Azure AI Search

Define artificial intelligence

Artificial Intelligence (AI) is increasingly prevalent in the software applications we use every day; including digital assistants in our homes and cellphones, automotive technology in the vehicles that take us to work, and smart productivity applications that help us do our jobs when we get there.

So what actually is artificial intelligence?

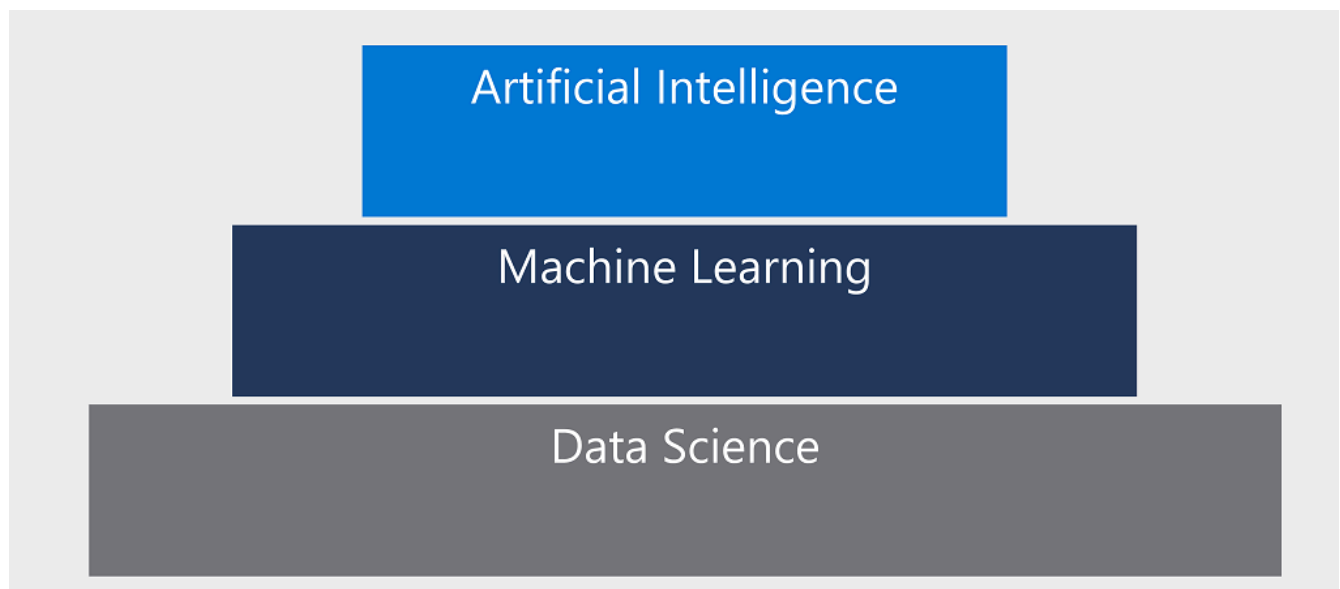
There are many definitions; some technical, some philosophical; but in general terms, we tend to think of AI as software that exhibits one or more human-like capabilities, as shown in the following table:

Capability	Description
	Visual perception - The ability to use <i>computer vision</i> capabilities to accept, interpret, and process input from images, video streams, and live cameras.
	Text analysis and conversation - The ability to use <i>natural language processing (NLP)</i> to not only "read", but also generate realistic responses and extract semantic meaning from text.
	Speech - The ability to recognize speech as input and synthesize spoken output. The combination of speech capabilities together with the ability to apply NLP analysis of text enables a form of human-compute interaction that's become known as <i>conversational AI</i> , in which users can interact with AI agents (usually referred to as <i>bots</i>) in much the same way they would with another human.
	Decision making - The ability to use past experience and learned correlations to assess situations and take appropriate actions. For example, recognizing anomalies in sensor readings and taking automated action to prevent failure or system damage.

These kinds of capabilities are increasingly within the reach of everyday software applications, helping make them more intuitive and useful in a wide variety of scenarios that previously existed only in the realms of science fiction.

Understand AI-related terms

There are several related terms that people use when talking about artificial intelligence, so it's useful to have clear definitions for each.



Data science

Data science is a discipline that focuses on the processing and analysis of data; applying statistical techniques to uncover and visualize relationships and patterns in the data, and defining experimental *models* that help explore those patterns.

For example, a data scientist might gather samples of data about the population of an endangered species in a geographical area, and combine it with data about levels of industrialization and economic demographics in the same area.

The data can then be analyzed, using statistical techniques to extrapolate from the samples to understand trends and relationships between human activities and wildlife, and test hypotheses using models that show the likely impact of human activity on the wildlife population.

By doing so, the data scientists may help determine optimal policies that balance the need for economic wellbeing for the human population with the need for conservation of the endangered wildlife.

Machine learning

Machine learning is a subset of data science that deals with the training and validation of *predictive* models. Typically, a data scientist prepares the data and then uses it to train a model based on an algorithm that exploits the relationships between the *features* in the data to predict values for unknown *labels*.

For example, a data scientist might use the data they have collected to train a model that predicts the annual growth or decline in population of a species based on factors such as the number of nesting sites observed, the area of land designated as protected, the human population in the local area, the daily volume of traffic on local roads, and so on. This predictive model can then be used as a tool to evaluate plans for housing, infrastructure, and industrial development in the local area and assess their likely impact on the local wildlife.

Artificial intelligence

Artificial intelligence usually (but not always) builds on machine learning to create software that **emulates one or more characteristics of human intelligence**.

For example, balancing the need for wildlife conservation against economic development requires accurate monitoring of the population of the endangered species being protected. It may not be feasible to rely on human experts who can positively identify the animal in question, or to monitor a large area over a sufficient period of time to get an accurate count. Indeed, the presence of human observers may deter animals and prevent their detection.

In this case, a predictive model could be trained to analyze image data taken by motion-activated cameras in remote locations, and predict whether a photograph contains a sighting of the animal. The model could then be used in a software application that responds to automated identification of animals to track animal sightings across a large geographical area, identifying areas with dense animal populations that may be candidates for protected status.

Understand considerations for AI Engineers

Increasingly, software solutions include AI features; so software engineers need to know how to integrate AI capabilities into their applications and services.

The advances made in machine learning, together with the increased availability of large volumes of data and powerful compute on which to process it and train predictive models, has led to the **availability of prepackaged software services that encapsulate AI capabilities**.

Software engineers can take advantage of these services to create applications and agents that use the **underlying AI functionality, using them as building blocks to create intelligent solutions**.

This means that software engineers can apply their existing skills in programming, testing, working with source control systems, and packaging applications for deployment, without having to become data scientists or machine learning experts.

However, to fully capitalize on the opportunities of AI, software engineers do require at least a conceptual understanding of core AI and machine learning principles.

Model training and inferencing

Many AI systems rely on predictive models that must be *trained* using sample data. The training process analyzes the data and determines relationships between the *features* in the data (the data values that will generally be present in new observations) and the *label* (the value that the model is being trained to predict).

After the model has been trained, you can submit new data that includes known *feature* values and have the model predict the most likely *label*. Using the model to make predictions is referred to as *inferencing*.

Many of the services and frameworks that software engineers can use to build AI-enabled solutions require a development process that involves training a model from existing data before it can be used to inference new values in an application.

Probability and confidence scores

A well-trained machine learning model can be accurate, but no predictive model is infallible. The predictions made by machine learning models are based on *probability*, and while software engineers don't require a deep mathematical understanding of probability theory, it's important to understand that predictions reflect statistical likelihood, not absolute truth. In most cases, predictions have an associated *confidence score* that reflects the probability on which the prediction is being made.

Software developers should make use of confidence score values to evaluate predictions and apply appropriate thresholds to optimize application reliability and mitigate the risk of predictions that may be made based on marginal probabilities.

Responsible AI and ethics

It's important for software engineers to consider the impact of their software on users, and society in general; including ethical considerations about its use. When the application is imbued with artificial intelligence, these considerations are particularly important due to the nature of how AI systems work and inform decisions; often based on probabilistic models, which are in turn dependent on the data with which they were trained.

The human-like nature of AI solutions is a significant benefit in making applications user-friendly, but it can also lead users to place a great deal of trust in the application's ability to make correct decisions.

The potential for harm to individuals or groups through incorrect predictions or misuse of AI capabilities is a major concern, and software engineers building AI-enabled solutions should apply due consideration to mitigate risks and ensure fairness, reliability, and adequate protection from harm or discrimination.

Understand considerations for responsible AI

The previous unit introduced the need for considerations for responsible and ethical development of AI-enabled software. In this unit, we'll discuss some core principles for responsible AI that have been adopted at Microsoft.

Fairness



AI systems should treat all people fairly. For example, suppose you create a machine learning model to support a loan approval application for a bank. The model should **make predictions of whether or not the loan should be approved without incorporating any bias based on gender, ethnicity, or other factors that might result in an unfair advantage or disadvantage to specific groups of applicants.**

Fairness of machine learned systems is a highly active area of ongoing research, and some software solutions exist for evaluating, quantifying, and mitigating unfairness in machine learned models.

However, tooling alone isn't sufficient to ensure fairness. Consider fairness from the beginning of the application development process; carefully reviewing training data to ensure it's representative of all potentially affected subjects, and evaluating predictive performance for subsections of your user population throughout the development lifecycle.

Reliability and safety



AI systems should perform reliably and safely. For example, consider an AI-based software system for an autonomous vehicle; or a machine learning model that diagnoses patient symptoms and recommends prescriptions. Unreliability in these kinds of system can result in substantial risk to human life.

As with any software, AI-based software application development must be subjected to rigorous testing and deployment management processes to ensure that they **work as expected before release**.

Additionally, software engineers need to take into account the **probabilistic nature of machine learning models, and apply appropriate thresholds when evaluating confidence scores for predictions**.

Privacy and security



AI systems should be secure and respect privacy. The machine learning models on which AI systems are based rely on large volumes of data, which may contain **personal details that must be kept private**.

Even after models are trained and the system is in production, they use new data to make predictions or take action that may be subject to privacy or security concerns; so appropriate safeguards to protect data and customer content must be implemented.

Inclusiveness



AI systems should empower everyone and engage people. AI should bring benefits to all parts of society, regardless of physical ability, gender, sexual orientation, ethnicity, or other factors.

One way to optimize for inclusiveness is to ensure that the design, development, and testing of your application includes **input from as diverse a group of people as possible**.

Transparency



AI systems should be understandable. Users should be made fully aware of the purpose of the system, how it works, and what limitations may be expected.

For example, when an AI system is based on a machine learning model, you should generally make users aware of factors that may affect the accuracy of its predictions, such as the number of cases used to train

the model, or the specific features that have the most influence over its predictions. You should also share information about the confidence score for predictions.

When an AI application relies on personal data, such as a facial recognition system that takes images of people to recognize them; you should make it clear to the user how their data is used and retained, and who has access to it.

Accountability



People should be accountable for AI systems. Although many AI systems seem to operate autonomously, ultimately it's the **responsibility of the developers who trained and validated the models they use, and defined the logic that bases decisions on model predictions to ensure that the overall system meets responsibility requirements.**

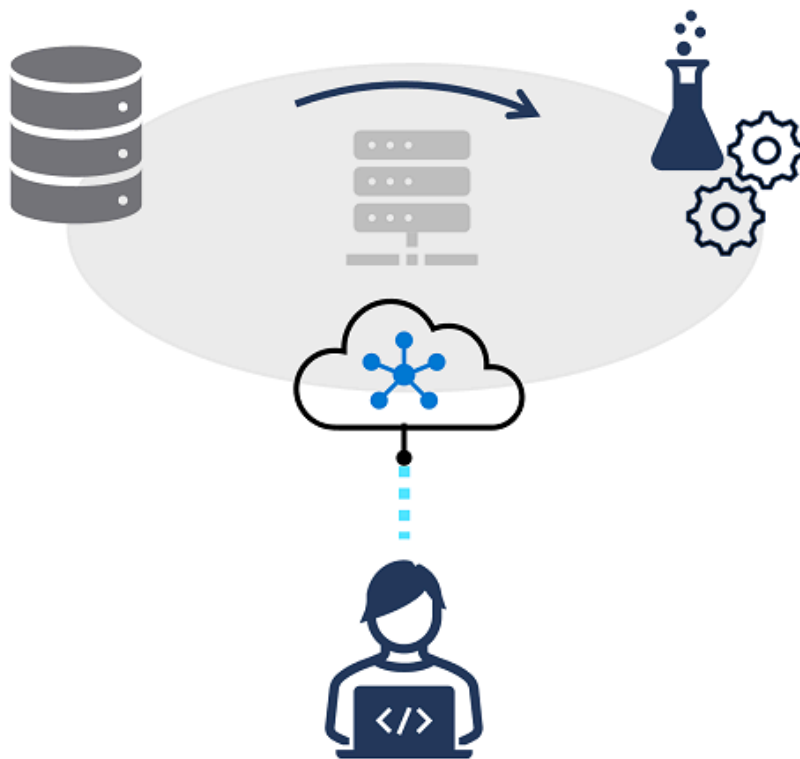
To help meet this goal, designers and developers of AI-based solution should work within a framework of governance and organizational principles that ensure the solution meets ethical and legal standards that are clearly defined.

Note: Microsoft has released [meaningful updates](#) to the Responsible AI Standard in June 2022. As part of that, we've updated the approach to facial recognition including a new Limited Access policy for certain features as a safeguard for responsible use. You can [apply for that limited access](#) to enable those features for your application. For more information about Microsoft's principles for responsible AI, visit [the Microsoft responsible AI site](#).

Understand capabilities of Azure Machine Learning

Microsoft Azure provides the **Azure Machine Learning** service - a cloud-based platform for **running experiments at scale to train predictive models from data, and publish the trained models as**

services.



Azure Machine Learning provides the following features and capabilities:

Feature	Capability
Automated machine learning	This feature enables non-experts to quickly create an effective machine learning model from data.
Azure Machine Learning designer	A graphical interface enabling no-code development of machine learning solutions.
Data and compute management	Cloud-based data storage and compute resources that professional data scientists can use to run data experiment code at scale.
Pipelines	Data scientists, software engineers, and IT operations professionals can define pipelines to orchestrate model training, deployment, and management tasks.

Data scientists can use Azure Machine Learning throughout the entire machine learning lifecycle to:

- Ingest and prepare data.
- Run experiments to explore data and train predictive models.
- Deploy and manage trained models as web services.

Software engineers may interact with Azure Machine Learning in the following ways:

- Using Automated Machine Learning or Azure Machine Learning designer to train machine learning models and deploy them as services that can be integrated into AI-enabled applications.
- Collaborating with data scientists to deploy models based on common frameworks such as Scikit-Learn, PyTorch, and TensorFlow as web services, and consume them in applications.
- Using Azure Machine Learning SDKs or command-line interface (CLI) scripts to orchestrate DevOps processes that manage versioning, deployment, and testing of machine learning models as part of an

overall application delivery solution.

For more information, see [Azure Machine Learning](#).

Understand capabilities of Azure AI Services

Azure AI Services are cloud-based services that encapsulate AI capabilities. Rather than a single product, you should think of **Azure AI Services as a set of individual services that you can use as building blocks** to compose sophisticated, intelligent applications.

Azure AI services offer a wide range of **pre-built AI capabilities** across multiple categories, with examples shown in the following table.

Natural language processing	Knowledge mining and document intelligence	Computer vision	Decision support	Generative AI
Text analysis	AI Search	Image analysis	Content safety	Azure OpenAI Service
Question answering	Document Intelligence	Video analysis	Content moderation	DALL-E image generation
Language understanding	Custom Document Intelligence	Image classification		
Translation	Custom skills	Object detection		
Named entity recognition		Facial analysis		
Custom text classification		Optical character recognition		
Speech		Azure AI Video Indexer		
Speech Translation				

For more information about Azure AI Services, see the [Azure AI Services](#) web page.

Understand capabilities of the Azure OpenAI Service

Generative AI is a relatively new and quickly progressing field of AI focused on AI models that *generate* content. Content that these models generate can be in the form of text, images, code or more, and in a way that almost feels like interacting with a real person in a real conversation. Generative AI models depend on *large language models* (LLMs) based on the transformer architecture that evolved from years of machine learning progress. Generative AI models are often queried with natural language prompts, and return an impressively accurate response when prompted correctly.

Azure OpenAI Service is an Azure AI service for deploying, utilizing, and fine-tuning models developed by OpenAI. OpenAI, the company who built ChatGPT, is one of the most popular applications most people have seen, and the **models behind that ChatGPT uses are available through the Azure OpenAI**

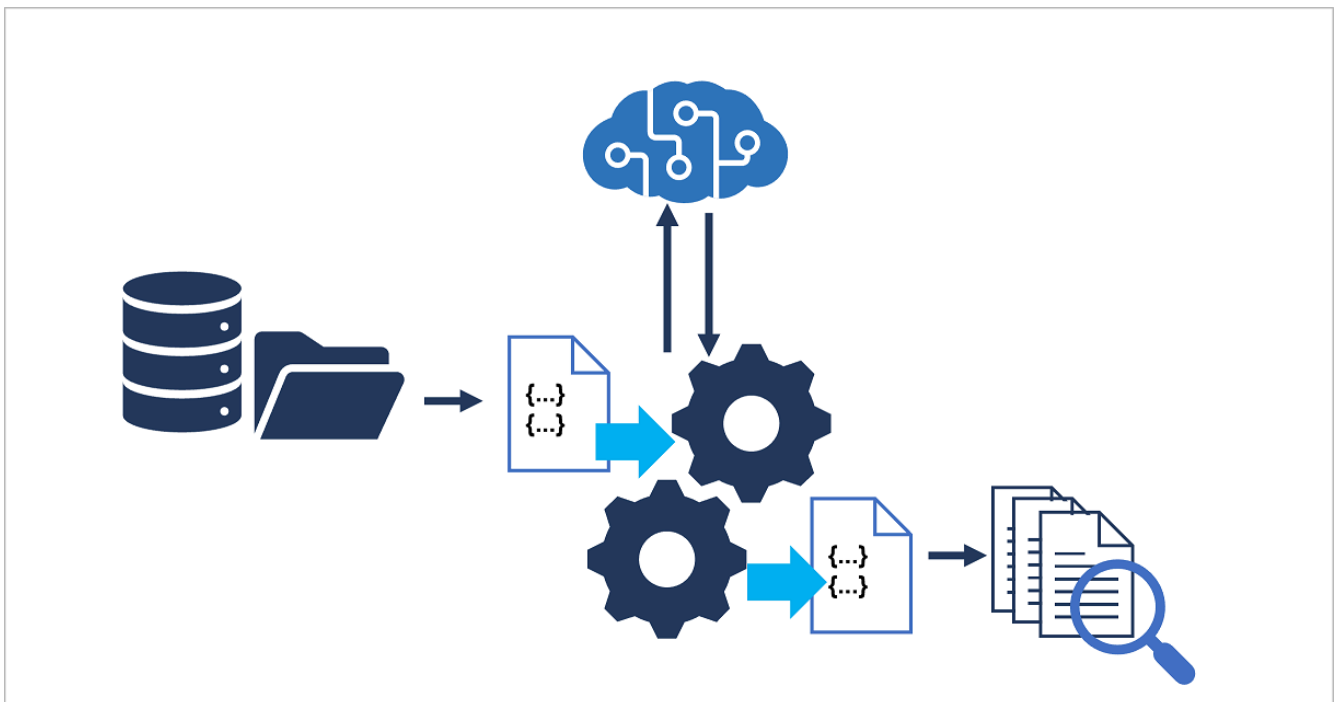
Service. You can develop applications that use the powerful generative AI models in Azure OpenAI to further utilize this technology.

AI engineers can develop applications that use the powerful generative AI models in Azure OpenAI to further utilize this technology. Both REST and language specific SDKs are available when developing applications.

For more information about Azure OpenAI and generative AI, see [the Azure OpenAI Service page](#).

Understand capabilities of Azure AI Search

Searching for information is a common requirement in many applications, from dedicated *search engine* web sites to mobile apps that can find context-appropriate information based on where you are and what you want to accomplish.



Azure AI Search is an Applied AI Service that enables you to ingest and index data from various sources, and search the index to find, filter, and sort information extracted from the source data.

In addition to **basic text-based indexing**, Azure AI Search enables you to define an *enrichment pipeline* that uses AI skills to **enhance the index** with insights derived from the source data - for example, by **using computer vision and natural language processing capabilities to generate descriptions of images, extract text from scanned documents, and determine key phrases in large documents that encapsulate their key points.**

Not only does this AI enrichment produce a more useful search experience, the insights extracted by your enrichment pipeline can be persisted in a *knowledge store* for further analysis or integration into a data pipeline for a business intelligence solution.

To learn more, see the [Azure AI Search page](#).

Knowledge Check

1. Which of the following best describes the predictions made by a machine learning model? *

- ☐ Absolutely correct values based on conditional logic.
- ☐ Randomly selected values with an equal chance of selection.
- ☒ Probabilistic values based on correlations found in training data.

2. A data scientist has used Azure Machine Learning to train a machine learning model. How can you use the model in your application? *

- ☒ Use Azure Machine Learning to publish the model as a web service.
- ☐ Export the model as an Azure AI service.
- ☐ You must build your application using the Azure Machine Learning designer.

3. You want to index a collection of text documents, and search them from a mobile application. Which service should you use to create the index? *

- ☐ The Azure AI service
- ☒ Azure AI Search
- ☐ Azure OpenAI Service

Summary

In this module, you learned how to:

- Define artificial intelligence
- Understand AI-related terms
- Understand considerations for AI Engineers
- Understand considerations for responsible AI
- Understand capabilities of Azure Machine Learning
- Understand capabilities of Azure AI Services
- Understand capabilities of Azure OpenAI Service
- Understand capabilities of Azure AI Search