

Describe fundamental principles of machine learning on Azure

The World of Machine Learning - Why It Matters

Why even consider Machine Learning?

Traditionally when building software applications, we write lines of code that would implement the various application functions.

Here we tell the computer what to do. But most of the times, application code tends to become complex in nature and difficult to manage.

With Machine Learning we can teach a computer to perform a task using data.

Real-World Use Cases of Machine Learning



We can use Machine Learning models to detect Spam email. Instead of writing rules to detect the different types of email that could be Spam, we feed in existing labeled spam emails. We then train the model based on the existing data.



Models can be used to detect fraudulent banking transactions.



Machine Learning enhances medical diagnosis by spotting early hidden patterns or anomalies in scans and records, which can help to detect diseases like cancer or heart conditions earlier and more accurately.

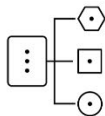
When should we consider using Machine Learning



When you have a large data set that you can feed into your model to train it. This is historical data that would already have labels.



When developing the code to solve the business problem is too difficult. And creating a Machine Learning model would prove to be more effective.



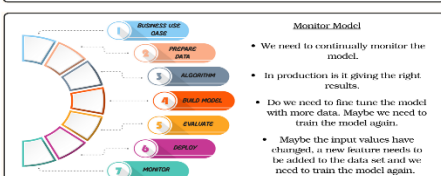
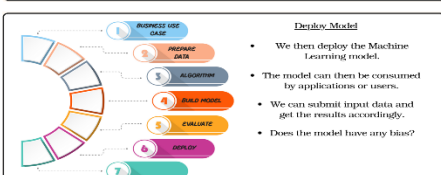
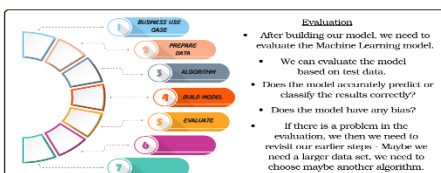
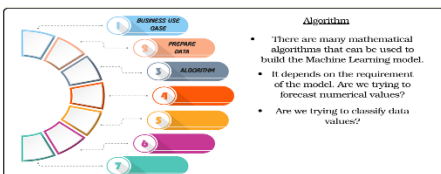
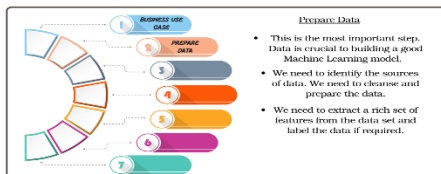
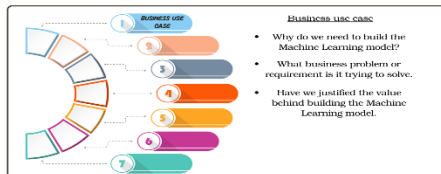
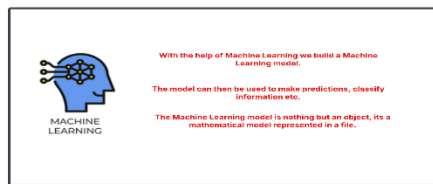
Classification

When the task at hand involves performing predictions, classifications, detections - Basically the core capabilities of a Machine Learning Model.

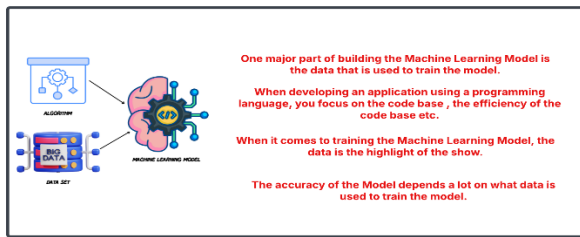
What is Machine Learning model - This is a mathematical representation that has already been trained on existing data. It looks at the data, discovers patterns and then can make decisions or predictions on new incoming data.

In the end the model is a file or a set of files. The files contain the structure of the model, the learned parameters like the weights and biases, any sort of preprocessing steps.

From Idea to Model - The ML Workflow

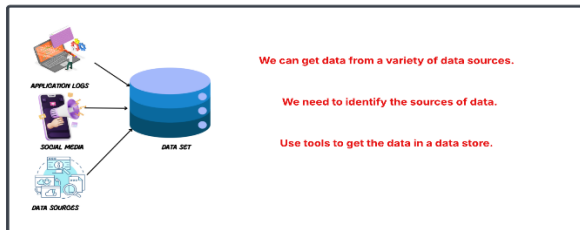


Your Data - Features, Labels & Splits

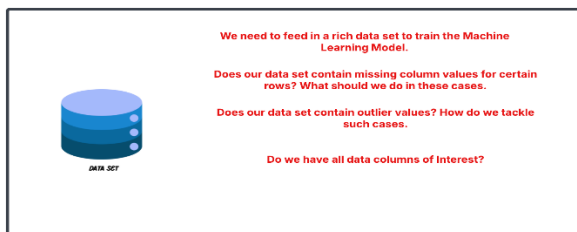


One of the main reasons for the popularity of Machine Learning is the vast amounts of data that is available for training a Machine Learning Model.

Phase 1 - First identify the different data sources



Phase 2 - Make sure the data is usable



Phase 3 - Generate a feature set

Location	Number of rooms	Agent	Area (sq ft)	Price (USD)
LocationA	5	AgentA	10,000	900,000
LocationB	4	AgentB	12,000	850,000

Now from the data set, we need to extract or generate features that can be fed into our Machine Learning Model.

The Machine Learning algorithm which is mathematical in nature will look at patterns in the feature set and then use those patterns to predict future values.

Feature - This is a measurable property within the data set.

Label - This is used to signify to the Machine Learning Model, that for a set of input values, this is the output value. In the end this is the value it needs to predict. This is used for Supervised Learning.

Feature Engineering - Generate new features based on existing ones.

The features need to make sense to the Machine Learning algorithm. Make sure the features are selected on the business problem that it is being targeted to solve.

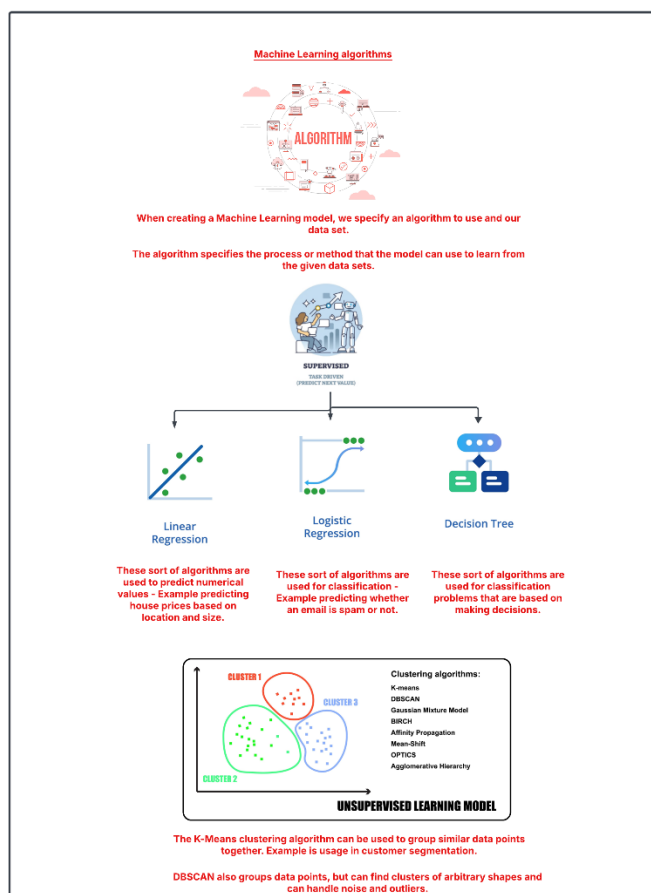
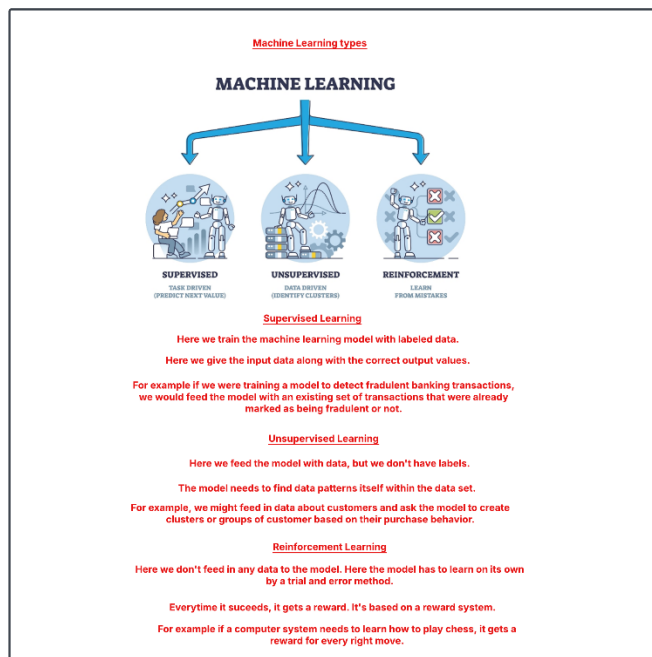
We also need to make sure we have a good spread of values in our data set that is used to train the model. There should be no bias in the data.

Let's say we are building a banking solution that can be used to detect fraudulent transactions.

We build a data set that has a historical set of transactions, a large data set. This data set has already been labelled. So under which conditions, parameters, features, the transaction has been detected as fraudulent.

Now if we feed in a data set that has 90% of transactions marked as non-fraudulent, the model has been trained very well on detecting non-fraudulent transactions. So even when making predictions in the real-world, it might classify actual fraudulent as non-fraudulent transactions.

Learning Styles - Supervised, Unsupervised & Beyond



Lab - Building Your ML Home - Creating a Workspace

The Azure Machine Learning workspace

This is an encapsulation of your machine learning artifacts that includes your data sets, your machine learning models etc.

When you deploy an Azure Machine Learning workspace , there are some Azure resources that get deployed along with it.



Storage Accounts

Azure storage account - When you run jobs to train your machine learning model, the logs are stored in the Azure storage account.



Container Registries

Azure Container Registry - This is used to store the images for created docker containers.



Application Insights

Azure Application Insights - This is used to collect diagnostic information from the inference endpoints.

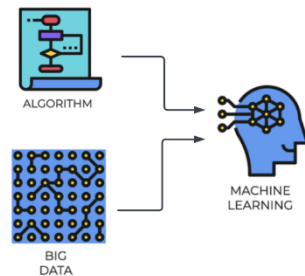


Key Vaults

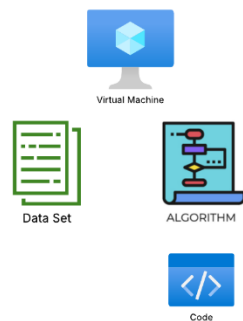
This is used to store secrets that might be needed by the workspace or other resources.

Lab - Binary Classification Model - Setting Up Your Compute Instance

Compute instance to run our pipeline



When you want to start training your machine learning model with training data and the machine learning algorithm, you need to have compute power in place.



Here remember that the algorithm is a set of instructions like code. These instructions need to go through the large data sets to look at patterns in data. Hence we need a compute machine to run the instructions and build our own machine learning model.

We can build a compute instance to run as part of our machine learning workspace to run the pipeline to build our machine learning model.

You have support for different target environments

Training targets	Automated machine learning	Machine learning pipelines	Azure Machine Learning designer
Local computer	Yes		
Azure Machine Learning compute cluster	Yes	Yes	Yes
Azure Machine Learning serverless compute	Yes	Yes	Yes
Azure Machine Learning compute instance	Yes (through SDK)	Yes	Yes
Azure Machine Learning Kubernetes		Yes	Yes
Remote VM	Yes	Yes	
Apache Spark pools (preview)	Yes (SDK local mode only)	Yes	
Azure Databricks	Yes (SDK local mode only)	Yes	
Azure Data Lake Analytics		Yes	
Azure HDInsight		Yes	
Azure Batch		Yes	

Reference - <https://learn.microsoft.com/en-us/azure/machine-learning/concept-compute-target?view=azureml-api-2>

Lab - Binary Classification Model - Evaluating the Model's Performance

The evaluation results depends on the type of Machine Learning algorithm being used to train the model.

Two-class Logistic regression

Confusion Matrix

With the Two-Class-Logistic Regression algorithm, we are doing binary classification where we just predict between two classes.

The Confusion matrix is a table that shows how well the model performed when comparing predicted with actual values.

		Actual	
		>50K	<=50K
Predicted	>50K	1 049	380
	<=50K	1 198	6 588

True Positive (TP) - Here the model predicted the value as positive and it was positive.

True Negative (TN) - Here the model predicted the value as negative and it was negative.

False Positive (FP) - Here the model predicted the value as positive but it was actually negative.

False Negative (FN) - Here the model predicted the value as negative but it was actually positive.

With the help of the matrix we can calculate important performance metrics.

Accuracy - This tells how often the classifier is right in predicting results.

$$\frac{\text{True Positives} + \text{True Negatives}}{\text{True Positives} + \text{True Negatives} + \text{False Positives} + \text{False Negatives}}$$

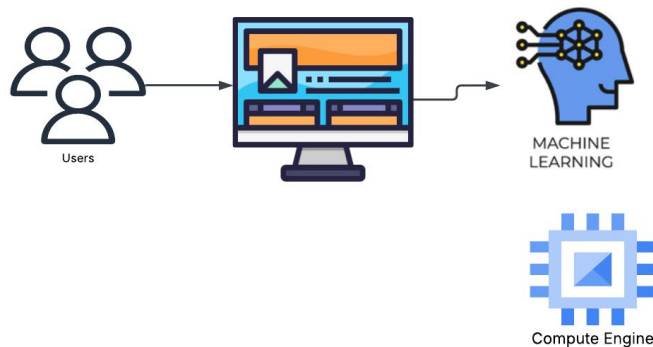
Precision - This tells to what extent does the model accurately predict results.

$$\frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Lab - Binary Classification Model - Creating a Kubernetes Cluster

Now we have built our machine learning model.
Let's say that we are happy with the accuracy and precision of the model.

We now want users to be able to submit requests to the model, specify input values and get a result back when it comes to the income.



Now we could embed the machine learning model within an application. So users could submit requests via the application, specify the input data and then get the results from the machine learning model.

One thing we need to understand is that the machine learning model needs to run on compute infrastructure. This is because the model needs to take in the input data, perform the required processing and give the output.

We also need to have an endpoint in place for the model. This is so that we can call the endpoint and hence make a call to the model.

From a successful run job, we will now create a real-time inference pipeline.

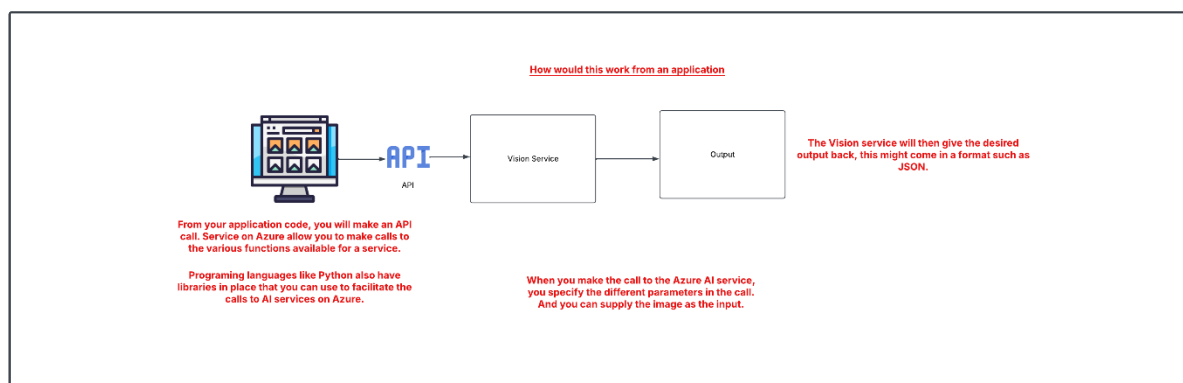
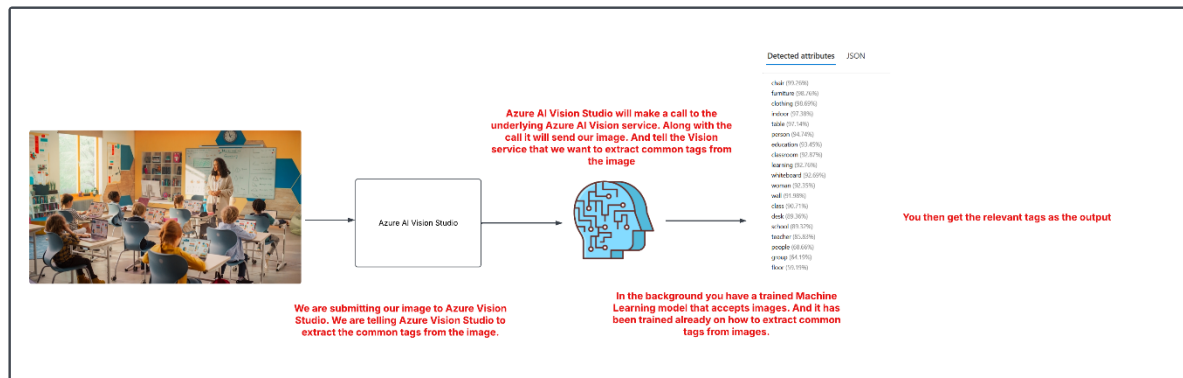
Once we have our real time inference endpoint in place, we can deploy this to a compute setup. Then we can call the endpoint.



We will create a new compute environment in Azure ML, which is a Kubernetes compute setup. The Machine Learning model would be then be hosted on this setup.

Describe features of computer vision workloads on Azure

Behind the Scenes - How Azure AI Vision Works



Describe features of generative AI workloads on Azure

Introduction to Generative AI

Generative Artificial Intelligence

We have seen how we can leverage AI-based services on Azure to identify objects, classify objects, detect faces, translate text etc.



OBJECT
RECOGNITION



IMAGE
RECOGNITION



FACE
DETECTION

In the backend you will have a trained machine learning model that could provide all of this functionality.

Generative AI is where the model can generate or produce output such as text, images, videos etc.

You can build machine learning models that can predict future values based on historical data. Or use the input to detect certain aspects such as object detection etc.

But Gen AI is in a different league altogether.

Here I am asking ChatGPT for advise, probably similar to asking a mentor on what I should do next.

It's courteous like a normal human being, telling us that we are moving in the right direction.

I am currently learning for the AZ-900 exam. My aim is to build my portfolio with cloud-based services. Can you suggest me based on the market trends , what should be next in my learning path.

Memory updated

That's a great step forward! The AZ-900 exam is an excellent foundation for understanding Azure's core concepts. Based on current market trends, here's a suggested learning path to complement your cloud knowledge and expand your portfolio:

Next Steps After AZ-900:

1. Specialized Azure Certifications:
 - **Azure Administrator (AZ-104):** Dive into managing cloud resources, storage, virtual networks, and security.
 - **Azure Developer (AZ-204):** Ideal for your .NET background, focusing on developing and deploying Azure applications.
 - **Azure Solutions Architect (AZ-305):** For designing advanced cloud solutions.
2. DevOps Practices:
 - **Azure DevOps Engineer (AZ-400):** Learn CI/CD pipelines, infrastructure as code, and monitoring, which are highly in demand.
3. AI and Data Specializations:
 - **AI Engineer (AI-102):** If you're interested in building AI solutions.
 - **Data Engineer (DP-203):** For working with big data and analytics pipelines.

Skills to Add to Your Portfolio:

1. Infrastructure as Code (IaC):
 - Tools: Terraform, Bicep, and Azure Resource Manager (ARM) Templates.

And then its giving me a run down on the next steps.

This is all based on GenAI wherein it is generating text on the fly based on our input request.

Understanding Large Language Models

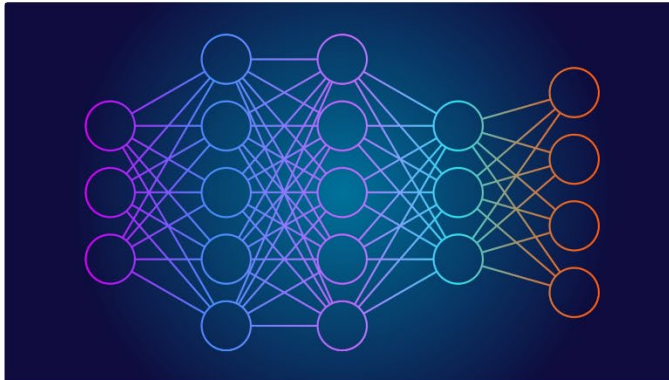
Large Language Models

This is a type of machine learning model which is specifically designed for tasks which involve natural language processing.

Here the model is trained on large amounts of data and are normally used to generate text.

We then have GPT - Generative pre-trained transformer.
This is a type of large-language model.

It is based on an artificial neural network.



This is a group of interconnected units called neurons.

It mimics the way our brain works. These neural networks are used to build artificial intelligence and make it behave similar to how we think and respond.

Its comprised of an input layer, hidden layers and an output layer. Each node or neuron in the network can use a simple or complex mathematical function to take the input and give the output. And the output of one neuron can then be given to the subsequent neuron.

ChatGPT is a generative artificial intelligence chatbot that was developed by a company known as OpenAI. Currently it is based on the GPT-4 large language model. The GPT, the transformer is what OpenAI developed to power ChatGPT.

Here the Chat part is just a friendly interface that allows us to interact with the Large Language model.

Now we have deployed simple machine learning models in our previous section. Can we build a large language model such as ChatGPT.

So the GPT models built by OpenAI has been trained on a large corpus of data - Think of the entire public data available on the Internet.

This large amount of data needs to be used in the training process. Then the outputs need to be validated. This is because, you take data from the Internet, the data can be wrong, have wrong intent or contain harmful content.

So you need to cater to all of this and retrain the model accordingly.

The initial version of GPT, GPT-1 was built on a parameter count of 117 million. And it was not disclosed to the public for fear for misuse.

The current version of GPT, GPT-4 is estimated to use around 1.7 trillion parameters.

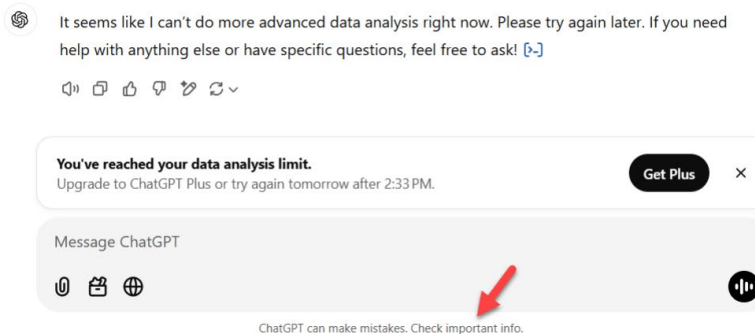
You need a tremendous amount of infrastructure to train and host the model. It runs into million and millions of dollars.

You need special GPU's that can handle the processing of data. You need a large number of the GPU's and its takes days and days to train the model.

Responsible AI - Using AI Ethically

We have seen how powerful ChatGPT is.

Nowadays a lot of individuals put a lot of dependency on the results generated by ChatGPT.



Next the GPT model was trained a lot of publicly available data. There is a lot of debate on whether it was right to take in all of the data.

What about data that its not trained on. If we ask it data about a recent event its not trained on, it should not propose a wrong output. This is also known as hallucination - This is wherein it tries to give us plausible answers that are incorrect.

With the data on the Internet, you can have data that biased in nature. Hence the model can also be biased. And this goes against the general principle of responsible AI.

Also ChatGPT has been trained to reject certain prompts that could violate certain policies. But users have found workarounds wherein it can ask ChatGPT with prompts that can violate this policy.

Working with GPT Models - Key Insights

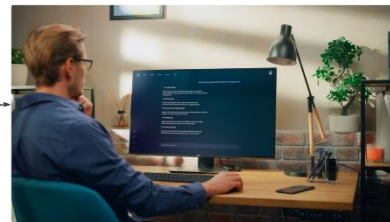
Let's have a review on some of the key concepts we have learnt so far.



OpenAI



OpenAI as a company has a lot of AI models on offer. An example is the GPT 4.1 model. Each model has its own set of capabilities.



Prompts

When interacting with a model, we have several ways of doing this. If we look at OpenAI, we can use the ChatGPT interface. Or we can issue API calls against the OpenAI platform.

The model uses its internal GPT engine to generate a response.

The response depends on how the model has been trained to understand various aspects such as the language, patterns and context.

Tokens

The model breaks the prompt text into tokens. These tokens are fed into the GPT model. The GPT then responds back again in the form of tokens. Tokens just symbolize the basic unit of text.

Multi-modal

Some models have the feature of being multi-modal in nature. Here the model has the capability to not only understand text as an input, but it can also understand submitted files, audio files etc.