Exam Pro

Azure AI-900 Cheat Sheets

Good luck on your exam!

AI-900 CheatSheet

Artificial Intelligence a machine that perform jobs that **mimic human behavior**
Machine Learning a machine that get better at a task **without explicit programming**
Deep Learning a machine that have an **artificial neural network** inspired by the human brain to solve complex problems.
Data Scientist a person with **multi-disciplinary skills** in math, statistics, predictive modeling and machine learning **to make future predictions**
dataset is a **logical grouping of units of data** that are closely related and/or share the same data structure

- eg. MNIST, COCO

Data Labeling the process of **identifying raw data** (images, text files, videos, etc.) **and adding one or more meaningful and informative labels** to **provide context** so that a machine learning model can learn
Supervised Learning (SL) Data that has been labeled for training
Unsupervised Learning (SL) Data has not been labeled, the ML model needs to do its own labeling
Reinforcement Learning (Rl) There is no data, there is an environment and an ML model generates data with many attempt to reach a goal
Neural Networks (NN) A network of nodes organized into layers (input, hidden, output) that is used to train ML models.
Deep Neural Network (DNN) A neural network that has **3 or more hidden layers** is considered deep learning.
Backpropagation (BP) Moves backwards through the neural net adjusting weights to improve outcome on next iteration. how a NN learns
Loss Function A function that compares the ground truth to the prediction to determine the error rate (how bad the network performed)
Activation Functions An algorithm applied to a hidden layer node that affects connected output e.g. ReLu
Dense Layer When the next layer increases the amount of nodes
Sparse Layer When the next layer decreases the amount of nodes
General Processing Unit (GPU) that is specially designed to quickly render high-resolution images and video **concurrently.**

- commonly used for **non-graphical tasks such as machine learning** and scientific computation.

Compute Unified Device Architecture (CUDA) is a **parallel computing platform** and **API** by NVIDIA that allows developers to use **CUDA-enabled GPUs** for general-purpose computing on GPUs (GPGPU)

AI-900 CheatSheet

ML Pipeline
- Pre Processing –preparing data and feature engineering before passing data to an ML model for training or inference
  - Data cleaning – correcting errors within the dataset that could negatively impact the results
  - Data reduction - reducing the amount of data, or applying dimensionality reductions to reduce the dimensions of inputted vectors
  - Feature engineering — transforming data in numerical (vectors) to be ingested by an ML model
  - Sampling – balancing a dataset to be uniform across labels by adding or removing records
- Post Processing – translating the output of an ML model back into a human readable format
- Training — the process of training the model
- Serving — the the process of deploying a model to an endpoint to be used for inference
- Inference — Invoking an ML model by sending a request and expecting back a prediction
  - Real-time endpoint – optimized for small or single item payloads, returns results quickly (a dedicated running server)
  - Batch-transform endpoint – optimized for larger batch predictions (server runs only for the duration of the batch)

Forecasting: Makes a future prediction with **relevant data,** analysis of trends, Its not "guessing"
Predicting: Makes a future prediction **without relevant data,** uses statistics to predict future outcomes, more of "guessing", Uses decision theory
Performance/Evaluation Metrics are used to evaluate different Machine Learning Algorithms
- Classification (Accuracy, F1 Score, Precision, Recall)
- Regression Metrics (MSE, RMSE MAE)

Jupyter Notebook A Web-based application for authoring documents that combine live-code , narrative text, equations, visualizations
Classification is a process of finding a function to **divide a labeled dataset into classes/categories**
- **Confusion matrix** is table to visualize the **model predictions** (predicted) vs **ground truth labels (actual)**

Regression is a process of finding a function to **correlate a labeled dataset into continuous variable/number**.
Clustering is a process **grouping unlabeled data based on similarities and differences**.

AI-900 CheatSheet

Cognitive Services is an umbrella AI service that enables customers to **access multiple AI services** with an **API key and an API Endpoint**

Decision

- **Anomaly Detector** — Identify potential problems early on.
- **Content Moderator —** Detect potentially offensive or unwanted content.
- **Personaliser —** Create rich, personalised experiences for every user.

Language

- **Language Understanding** — Build natural language understanding into apps, bots and IoT devices.
- **QnA Maker** — Create a conversational question and answer layer over your data.
- **Text Analytics** — Detect sentiment, key phrases and named entities.
- **Translator** — Detect and translate more than 90 supported languages.

Speech

- **Speech to Text** — Transcribe audible speech into readable, searchable text.
- **Text to Speech** — Convert text to lifelike speech for more natural interfaces.
- **Speech Translation** — Integrate real-time speech translation into your apps.
- **Speaker Recognition** — Identify and verify the people speaking based on audio.

Vision

- **Computer Vision** — Analyze content in images and video.
- **Custom Vision** — Customize image recognition to fit your business needs.
- **Face** — Detect and identify people and emotions in images.

Knowledge mining is a **discipline** in AI that uses a **combination of intelligent services to quickly learn from vast amounts of information.**

- **Ingest** content from a range of sources, using connectors to first and third-party data stores.
- **Enrich** the content with AI capabilities that let you extract information, find patterns, and deepen understanding.
- **Explore** the newly indexed data via search, bots, existing business applications, and data visualizations.

AI-900 CheatSheet

Micorsoft AI Principles (Responsible AI)

1. Fairness — AI systems should treat all people fairly
2. Reliability and Safety — AI systems should perform reliably and safely
3. Privacy and Security — AI systems should be secure and respect privacy
4. Inclusiveness — AI systems should empower everyone and engage people
5. Transparency — AI systems should be understandable
6. Accountability — People should be accountable for AI systems

Common ML Workloads:

- **Anomaly Detectio**n is the process of finding outliers within a dataset called an **anomaly**
- **Computer Vision** is when we use ML NN to gain high-level understanding from digital images or video
- **Natural Language Processing (NLP)** is ML that can understand the context of a corpus (a body of related text).
- **Conversational AI** is technology that can participate in conversations with humans.

AI-900 CheatSheet

Azure Machine Learning Service allows you to provision an ML studio to build and maintain ML models and pipelines

Author

- **Notebooks** — Jupyter Notebooks, an IDE to write python code to build ML models
- **AutoML** — Completely automated process to build and train an ML model
- **Designer** — Visual drag and drop designer to construct end to end ML pipelines

Assets

- **Datasets** — data that you upload which will be used for training. Datasets can be versioned
    - **Open DataSets** are **publicly hosted datasets** that are commonly **used for learning how to build ML models**
- **Experiments** — Experiments are logical grouping of runs.
    - **Runs** are ml tasks that performed on virtual machines or containers
- **Pipelines** — ML workflows you have built, or you have used in the Designer
    - Training Pipeline — pipelines to build and train an ml model
    - Inference Pipeline — pipelines that use a trained model to make prediction on real data.
- **Models** — a model registry containing trained models that can be deployed
- **Endpoints** — when you deploy a model its hosted on an accessible endpoint eg. REST API
    - Real-time Endpoint — Invokes an ML model for inference
    - Pipeline Endpoint — Invoke the running on a Pipeline eg. CI/CD

Manage

- **Compute —** the underlying computing instances used to for notebooks, training, inference
    1. **Compute Instances —** Development workstations that data scientists can use to work with data and models.
    2. **Compute Clusters —** Scalable clusters of virtual machines for on-demand processing of experiment code.
    3. **Inference Clusters —** Deployment targets for predictive services that use your trained models.
    4. **Attached Compute —** Links to existing Azure compute resources, such as Virtual Machines or Azure Databricks clusters.

AI-900 CheatSheet

- **Environments** — a reproducible Python environment for machine learning experiments

Datastores —**securely connect to your storage service** on Azure without **putting your authentication credentials**

  - Azure Blob Storage, Azure File Share, Azure Data Lake Storage (Gen 2), Azure SQL database, Azure Postgres/MySQL database

- **Data Labeling** — have humans with ML-assisted labeling to label your data for supervised learning
  - Human-in-the-loop labeling
  - Machine-learning-assisted data labeling
- **Linked Services** — external services you can connect to the workspace eg. Azure Synapse Analytics

Text Analytics

- sentiment analysis find out what people think of your brand or topic
  - Labels include **negative, positive, mixed or neutral**
  - Confidence scores ranging from 0 to 1
- opinion mining granular information about the opinions related to aspects
  - granular data with a **Subject** and **Opinion** tied to a Sentient
- key phrase extraction quickly identify the main concepts in text.
- language detection detect the language an input text is written in
- named entity recognition (NER) — detects **words and phrases mentioned in unstructured text** that can be **associated with one or more semantic types**.

AI-900 CheatSheet

Language Understanding (LUIS) is **a no-code ML service to build natural language into apps**, bots, and IoT devices

- Natural Language Understanding (NLU) the ability to *transform* a linguistic statement to a representation that enables you to understand your users naturally
- LUIS key schema compoennts
- **intentions** — what the user is asking for
  - a LUIS app always contains a **None** Intent
- **entities** — what parts of the intent is used to determine the answer
- **utterances** — Examples of user input that includes intent and entities to train the ML model to match predictions against real user input

QnA Maker generate a bot from a URL, PDF or DOCX

- multi-turn conversation – follow up prompts to narrow to a specific anwser
- Chit-chat — personalized canned responses

Azure Bot Service — allows you to host bots

- Bot Framework SDK — an end-to-end SDK to design, build, test, publish, connect and evaluate bots
- Box Framework Composer — a desktop application to design bots, leverages the Box Framework SDK

AI-900 CheatSheet

Artificial Intelligence (AI)
AI refers to the development of computer systems that can **perform tasks typically requiring human intelligence**. These include **problem-solving, decision-making, understanding natural language, recognizing speech and images**, and more.

Generative AI
Generative AI is a subset of AI that focuses on **creating new content or data** that is novel and realistic. It does not just interpret or analyze data but **generates new data itself**.

- It includes **generating text, images, music, speech, and other forms of media**.

A **Large Language Model (LLM)** is a type of artificial intelligence program that processes and generates human-like text by predicting subsequent parts of sentences based on massive amounts of data.

- It's designed to understand context, answer questions, complete tasks, and engage in conversations in a way that mimics natural human language.

A **transformer model** is a type of machine learning model that's especially good at **understanding and generating language**.
Transformer model architecture consists of **two components**, or *blocks*:
1. Encoder: This part **reads and understands the input text**. It's like a smart system that goes through everything it's been taught (which is a lot of text) and picks up on the meanings of words and how they're used in different contexts.
2. Decoder: Based on what the encoder has learned, this part **generates new pieces of text**. It's like a skilled writer that can make up sentences that flow well and make sense.
Tokenization in a transformer model is like turning a sentence into a puzzle. For example, you have the sentence: **"I heard a dog bark loudly at a cat."** To help a computer understand it, we chop up the sentence into pieces called **'tokens'**. Each piece can be a word or even a part of a word.

- To help a computer understand language, we turn words into tokens and then give each token a special **numeric code**, called an **embedding**. These embeddings are like a secret code that captures the meaning of the word.

AI-900 CheatSheet

Exam Pro

Positional encoding is a technique used to ensure that a language model, such as GPT (Generative Pre-trained Transformer) doesn't lose the **order of words** when processing natural language. This is important because the order in which words appear can change the meaning of a sentence.

In AI, **attention** is a mechanism that allows models, particularly in natural language processing, to focus on specific parts of the input data when performing a task, similar to how humans pay attention to certain aspects of what they see or hear.

- It helps the model prioritize which information to consider and which to ignore, improving its ability to understand context and make more accurate predictions.

A transformer model like **GPT-4** works by taking a text **input (prompt)** and producing a well-structured **output (completion)**. During training, it learns from a vast array of text data, understanding how words are typically arranged in sentences.

- The model knows the correct sequence of words but **hides (masks)** future words to learn how to **predict** them. When it tries to predict a word, it compares its guess to the actual word, gradually adjusting to reduce errors.

Azure OpenAI Service is a cloud-based platform designed to **deploy and manage advanced language models from OpenAI**. This service combines OpenAI's latest language model developments with the robust security and scalability of Azure's cloud infrastructure.

Azure OpenAI offers several **types of models** for different purposes:

GPT-4 Models: These are the newest in the line of GPT models and can create **text and programming code** when given a prompt written in natural language.

GPT-3.5 Models: Similar to GPT-4, these models also create text and code from natural language prompts. The GPT-3.5-turbo version is specially designed for **conversations**, making it a great choice for **chat applications and other interactive AI tasks**.

Embedding Models: These models turn written **text into number sequences**, which is helpful for analyzing and comparing different pieces of text to find out how **similar** they are.

DALL-E Models: These models can make **images from descriptions** given in words. The DALL-E models are still being tested and aren't shown in the Azure OpenAI Studio, so you don't have to set them up for use manually.

AI-900 CheatSheet

Exam Pro

Developers can work with these models in **Azure OpenAI Studio**, a **web-based environment** where AI professionals can **deploy, test, and manage LLMs** that support generative AI app development on Azure.
In **Azure AI Studio**, you can deploy **large language models**, provide few-shot examples, and **test** them in Azure OpenAI Studio's Chat playground.
Copilots are a new type of computing tool that integrates with applications to help users with **common tasks using generative AI models**. They are designed using a standard architecture, allowing developers to create **custom copilots** tailored to specific business needs and applications. Copilots might appear as a chat feature beside your document or file, and they **utilize the content within the product to generate specific results**.
Creating a copilot involves several steps:
1. Training a **large language model** with a vast amount of data.
2. Utilizing services like **Azure OpenAI Service**, which provide pre-trained models that developers can either use as-is or fine-tune with their own data for more specific tasks.
3. **Deploying** the model to make it available for use within applications.
4. **Building copilots** that **prompt** the models to generate usable content.
5. Business users can use copilots to boost their **productivity and creativity** with AI-generated content.

Microsoft Copilot is integrated into various applications to assist users in creating **documents, spreadsheets, presentations, and more**, by generating content, summarizing information, and aiding in strategic planning.
The **Microsoft Bing** search engine provides a copilot to help when **browsing or searching** the Internet by generating **natural language answers** to questions based on context rather than just search results of indexed pages.
Microsoft 365 Copilot is designed to be a partner in your workflow, integrated with productivity and communication tools like **PowerPoint** and **Outlook**.
GitHub Copilot is a tool that helps software developers, offering real-time assistance as they **write code**. It offers more than suggesting code snippets; it can help in **documenting the code** for better understanding and maintenance.

AI-900 CheatSheet

Exam Pro

Prompt engineering is a process that **improves the interaction between humans and generative AI**. It involves **refining the prompts** or instructions given to an AI application to **generate higher quality responses**. This process is valuable for both the developers who create AI-driven applications and the end-users who interact with them.

System messages

Prompt engineering techniques include defining a system message. The message sets the **context for the model** by describing **expectations** and **constraints**. For example, "You're a helpful assistant that responds in a **cheerful, friendly manner**".

These system messages determine constraints and styles for the model's responses.

Writing good prompts

To maximize the utility of AI responses, it is essential to be **precise and explicit** in your prompts.

A well-structured prompt, such as **"Create a list of 10 things to do in Edinburgh during August,"** directs the AI to produce a targeted and relevant output, achieving better results.

Zero-shot learning refers to an AI model's ability to correctly perform a task **without any prior examples or training** on that specific task.

One-shot learning involves the AI model learning from a **single example or instance** to perform a task.

Grounding in prompt engineering is a technique used in large language models (LLMs) where you provide **specific, relevant context within a prompt**. This helps the AI to produce a more accurate and related response.

The Microsoft guidance for **responsible generative AI** is designed to be practical and actionable. It defines a four stage process to develop and implement a plan for responsible AI when using generative models.

The four stages in the process are:

Identify: Recognize potential harms related to your AI solution.

Measure: Assess how often these harms occur in your AI outputs.

Mitigate: Reduce the harms and clearly communicate risks to users.

Operate: Follow a thorough plan for deployment and management.