

AI-102— Extra modules



+
o

Demo: Prompt engineering – med kode





Hugging Face

Hugging face og modeller

The screenshot shows the Hugging Face website's model library interface. At the top, there are tabs for Tasks, Libraries, Datasets, Languages, Licenses, and Other. A search bar labeled "Filter Tasks by name" is present. Below the search bar, there are sections for Multimodal (Image-Text-to-Text, Visual Question Answering, Document Question Answering), Computer Vision (Depth Estimation, Image Classification, Object Detection, Image Segmentation, Text-to-Image, Image-to-Text, Image-to-Image, Image-to-Video, Unconditional Image Generation, Video Classification, Text-to-Video, Zero-Shot Image Classification, Mask Generation, Zero-Shot Object Detection, Text-to-3D, Image-to-3D, Image Feature Extraction), and Natural Language Processing (Text Classification, Token Classification). Each category has several sub-options listed with small icons.

MODELS

Find the right model to build your custom AI solution

Show filters

Announcements

Mistral Small is now available!



Mistral AI's smallest yet highly efficient model, now available on Azure

[View models](#)

[Read blog](#)

Phi-3 is now available



Microsoft's Phi-3-mini SLMs offer groundbreaking performance at a small size

[View models](#)

[Read blog](#)

Build the future of AI with Meta Llama 3



Serverless APIs for Meta-Llama-3-8B-Instruct and Meta-Llama-3-70B-Instruct models

[View models](#)

[Read blog](#)

All filters

Collections

Inference tasks

Fine-tuning tasks

Licenses

Search

Models 1643



Chat completion



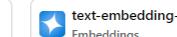
Text to image



Chat completion



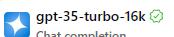
Completions



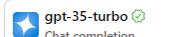
Embeddings



Chat completion



Chat completion



Chat completion



Completions



Chat completion



Text generation



Chat completion



Chat completion



Text generation



Text generation



Chat completion



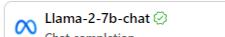
Text generation



Chat completion



Chat completion



Chat completion



Text generation



Text generation



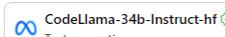
Text generation



Text generation



Text generation



Text generation



Text generation



Text generation



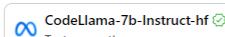
Chat completion



Text generation



Text generation



Text generation



Text generation



Text generation



Text generation



Chat completion

HOW TO USE MODELS

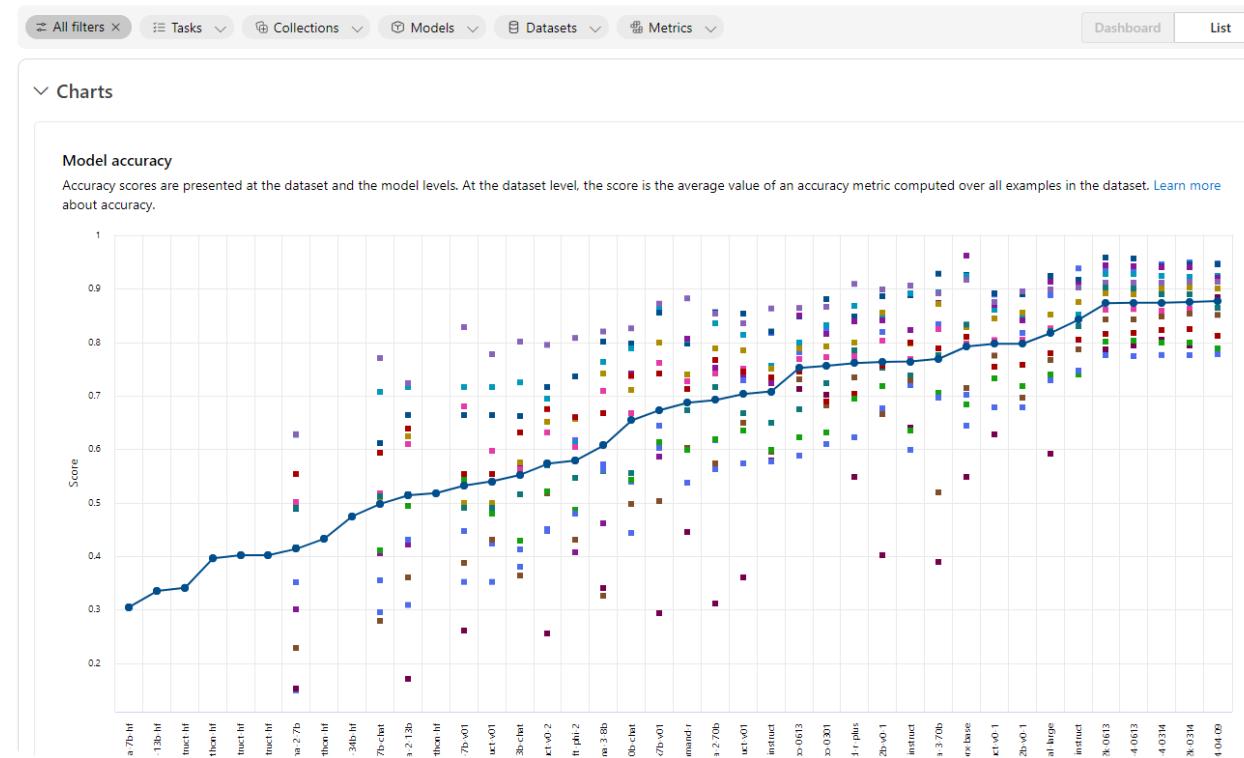
- **Discover:** Review model cards, try sample inference and browse code samples to evaluate, fine-tune, or deploy the model.
- **Compare:** Compare benchmarks across models and datasets available in the industry to assess which one meets your business scenario.
- **Evaluate:** Evaluate if the model is suited for your specific workload by providing your own test data. Evaluation metrics make it easy to visualize how well the selected model performed in your scenario.
- **Fine-tune:** Customize fine-tunable models using your own training data and pick the best model by comparing metrics across all your fine-tuning jobs. Built-in optimizations speedup fine-tuning and reduce the memory and compute needed for fine-tuning.
- **Deploy:** Deploy pretrained models or fine-tuned models seamlessly for inference. Models that can be deployed to real-time endpoints can also be downloaded.



BENCHMARKING MODELS

Assess model performance with evaluated metrics

Compare benchmarks across models and datasets available in the industry to assess which one meets your business scenario. Learn more about how model performance is scored



HuggingFace

- <https://huggingface.co/docs/transformers/quicktour>
- <https://huggingface.co/docs/transformers/index>



HUGGING FACE - DEMO



Hugging

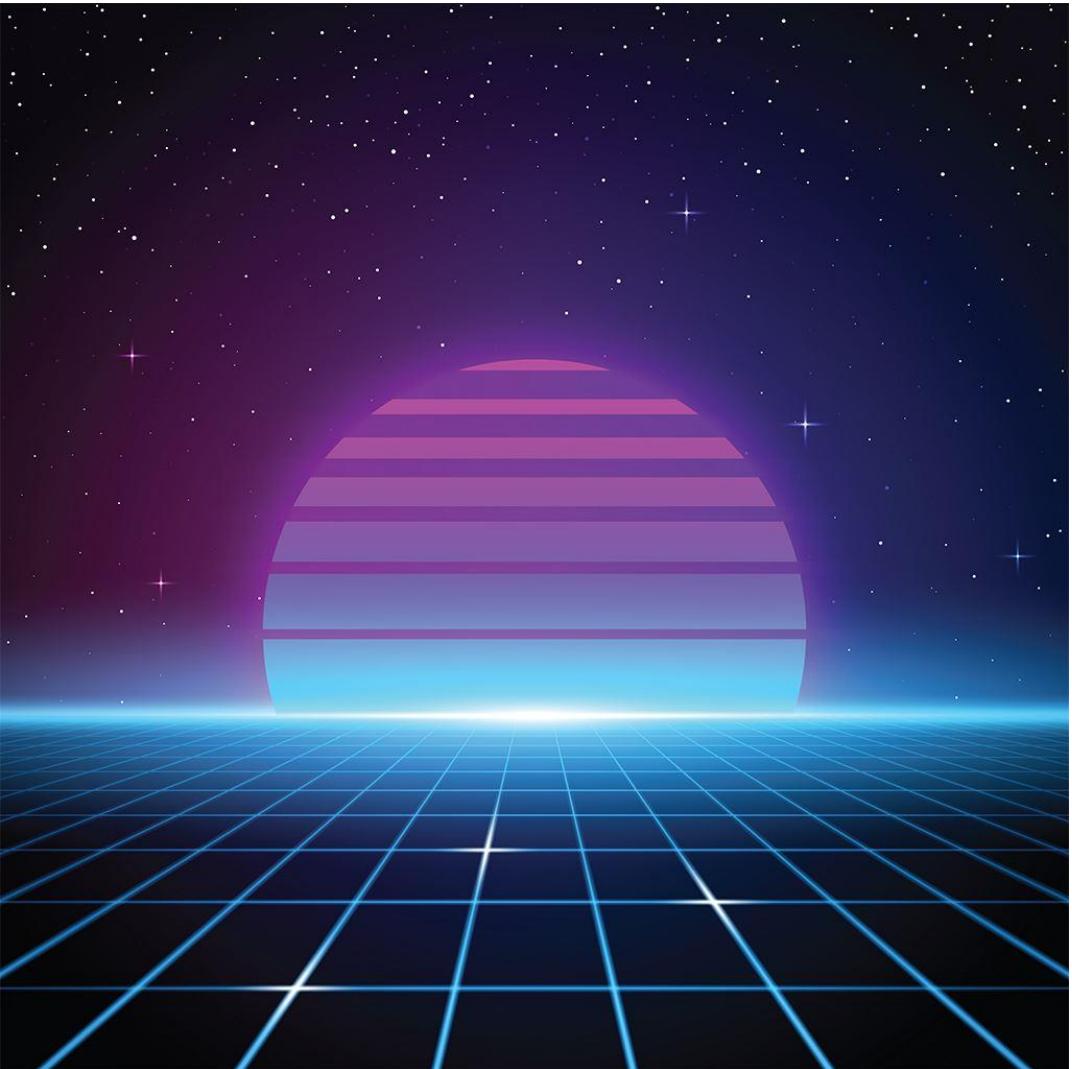
GENERATIV AI - EMBEDDINGS

MODUL 3



USING EMBEDDINGS WITH AZURE AI SEARCH AND PROMPTING

- Azure AI Search enables developers to discover and reuse existing AI solutions
- Embeddings are commonly used for search, clustering, recommendations, and classification
- Azure PromptFlow allows developers to coordinate individual parts of your AI solutions



EMBEDDINGS

Elements in a token embedding vector as coordinates in multidimensional space

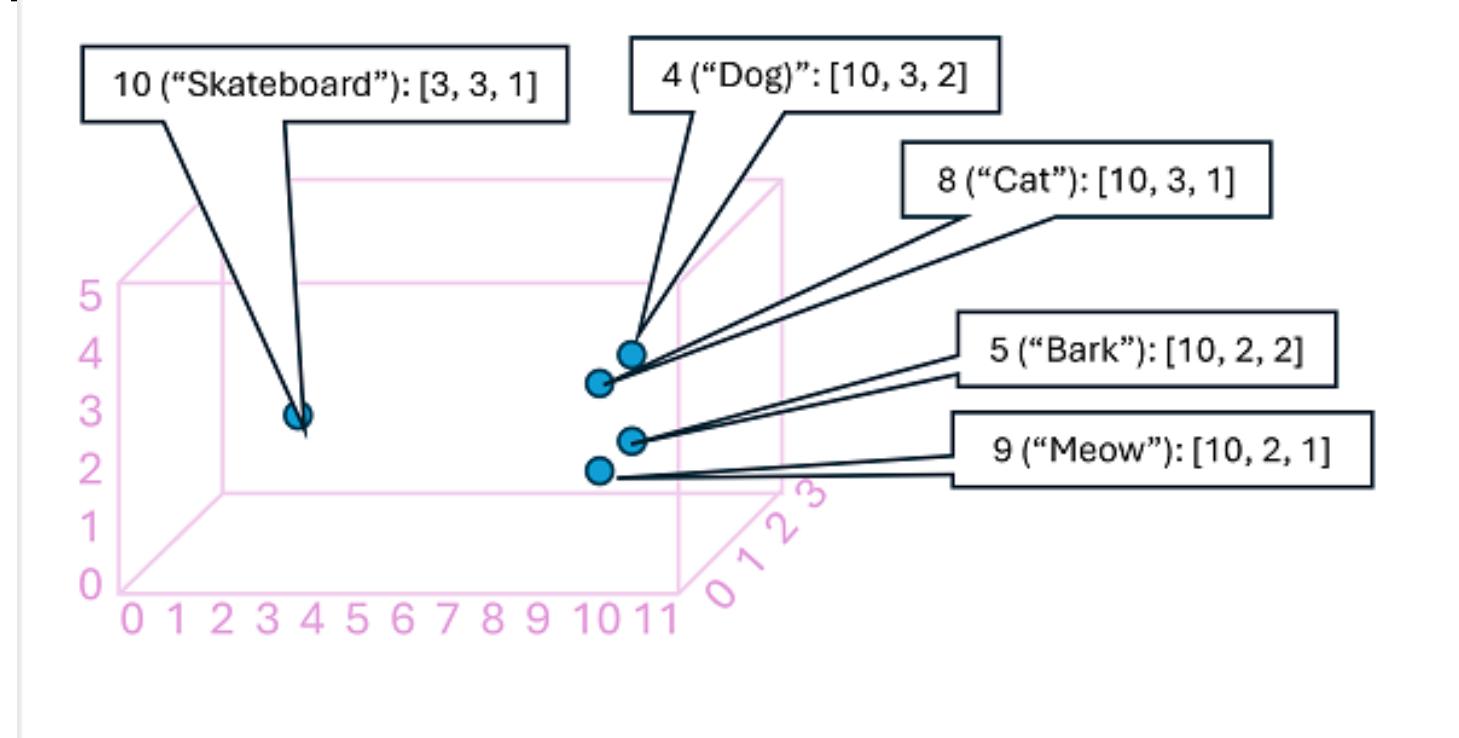
Each token occupies a specific "location."

The closer tokens are to one another along a particular dimension, the more semantically related they are.

In other words, related words are grouped closer together.

EMBEDDINGS - AN EXAMPLE

- 4 ("dog"): [10,3,2]
- 5 ("bark"): [10,2,2]
- 8 ("cat"): [10,3,1]
- 9 ("meow"): [10,2,1]
- 10 ("skateboard"):
[3,3,1]



EMBEDDING IN PRACTICE

1.Text Input: A user sends a message in the chat.

2.Azure OpenAI Service: The message is sent to the Azure OpenAI service.

3.Embedding Generation: Azure OpenAI converts the text message into an embedding, which is a vector of numbers representing the semantic meaning of the text.

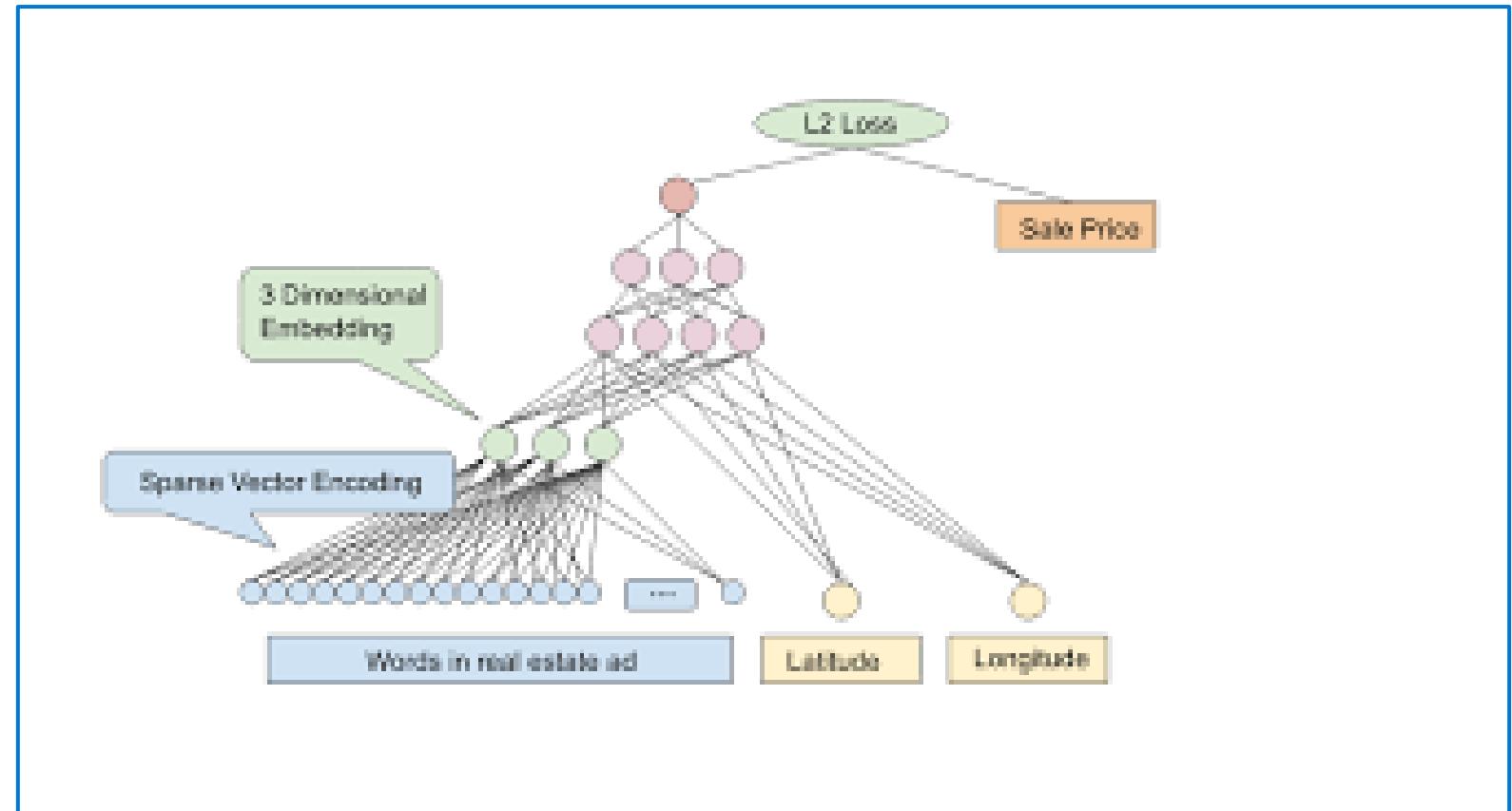
4.Semantic Search: The embedding is used to perform a semantic search in Azure AI Search.

5.Search Results: The most semantically relevant documents or information are retrieved based on the similarity of embeddings.

1. User: "How do I reset my password?"
2. Azure OpenAI Service: Receives the query and generates an embedding.
3. Embedding: [0.85, -0.23, 0.91, ..., 0.42]
// A vector representing the query.
4. Azure AI Search: Uses the embedding to find the most relevant documents.
5. Search Results: Returns documents related to password resetting.

Embedding

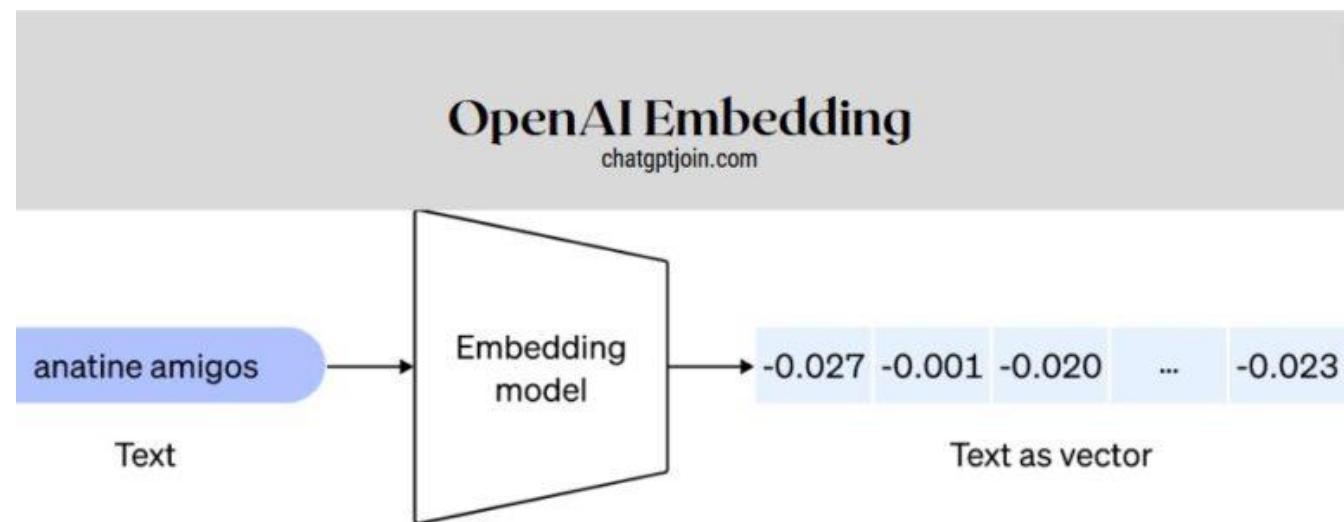
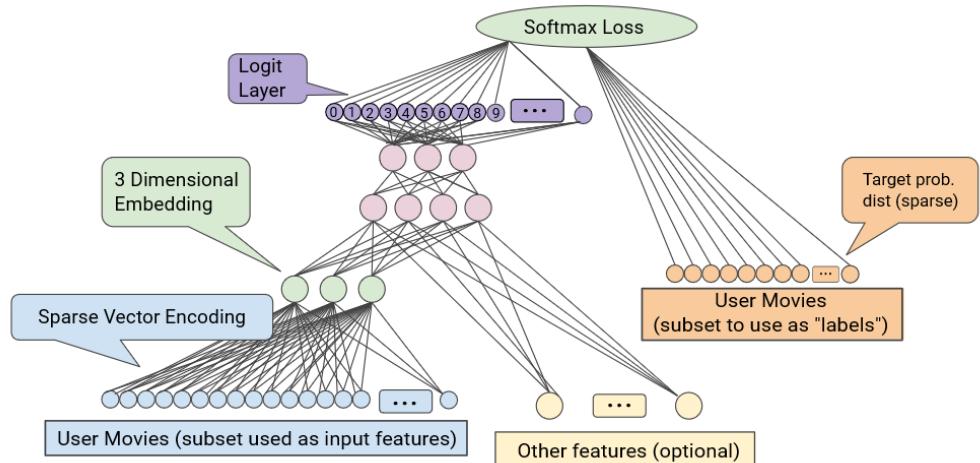
An embedding is a special format of data representation that can be easily utilized by machine learning models and algorithms. The embedding is an information dense representation of the semantic meaning of a piece of text. Each embedding is a vector of floating point numbers, such that the distance between two embeddings in the vector space is correlated with semantic similarity between two inputs in the original format.



https://ai.google.dev/docs/embeddings_guide

Embeddings

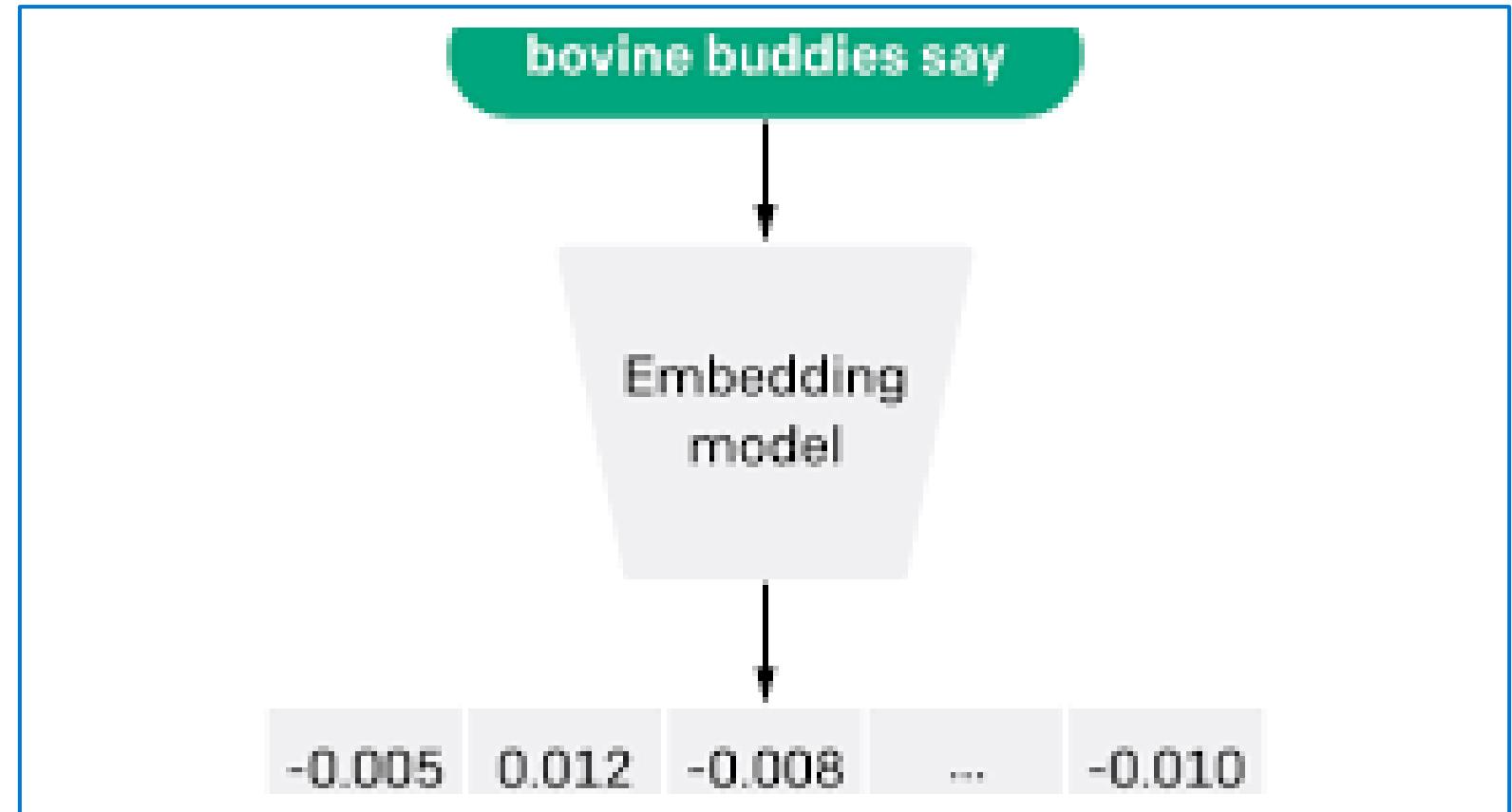
<https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/embeddings?tabs=python-new>



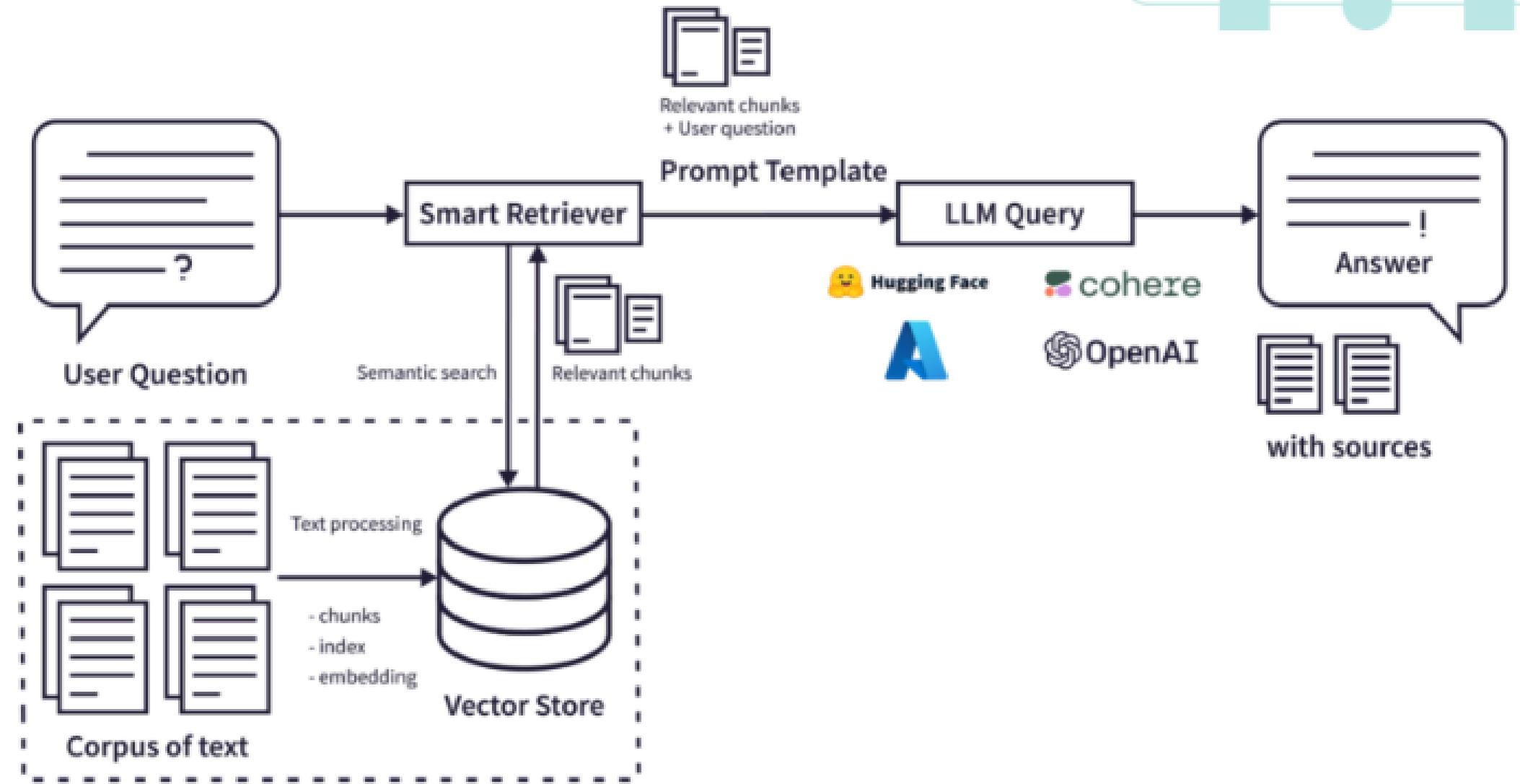
OpenAI Embedding Pricing: What You Need to Know

Embeddings – Demo/Lab

https://github.com/Azure-Samples/openai/blob/main/Basic_Samples/Embeddings/basic_embeddings_example_restapi.ipynb



RAG pipeline



EMBEDDINGS USE CASES

- What are embeddings?
- OpenAI's text embeddings measure the relatedness of text strings. Embeddings are commonly used for:
 - **Search** (where results are ranked by relevance to a query string)
 - **Clustering** (where text strings are grouped by similarity)
 - **Recommendations** (where items with related text strings are recommended)
 - **Anomaly detection** (where outliers with little relatedness are identified)
 - **Diversity measurement** (where similarity distributions are analyzed)
 - **Classification** (where text strings are classified by their most similar label)
- An embedding is a vector (list) of floating point numbers. The distance between two vectors measures their relatedness. Small distances suggest high relatedness and large distances suggest low relatedness.

EMBEDDINGS - DEMO



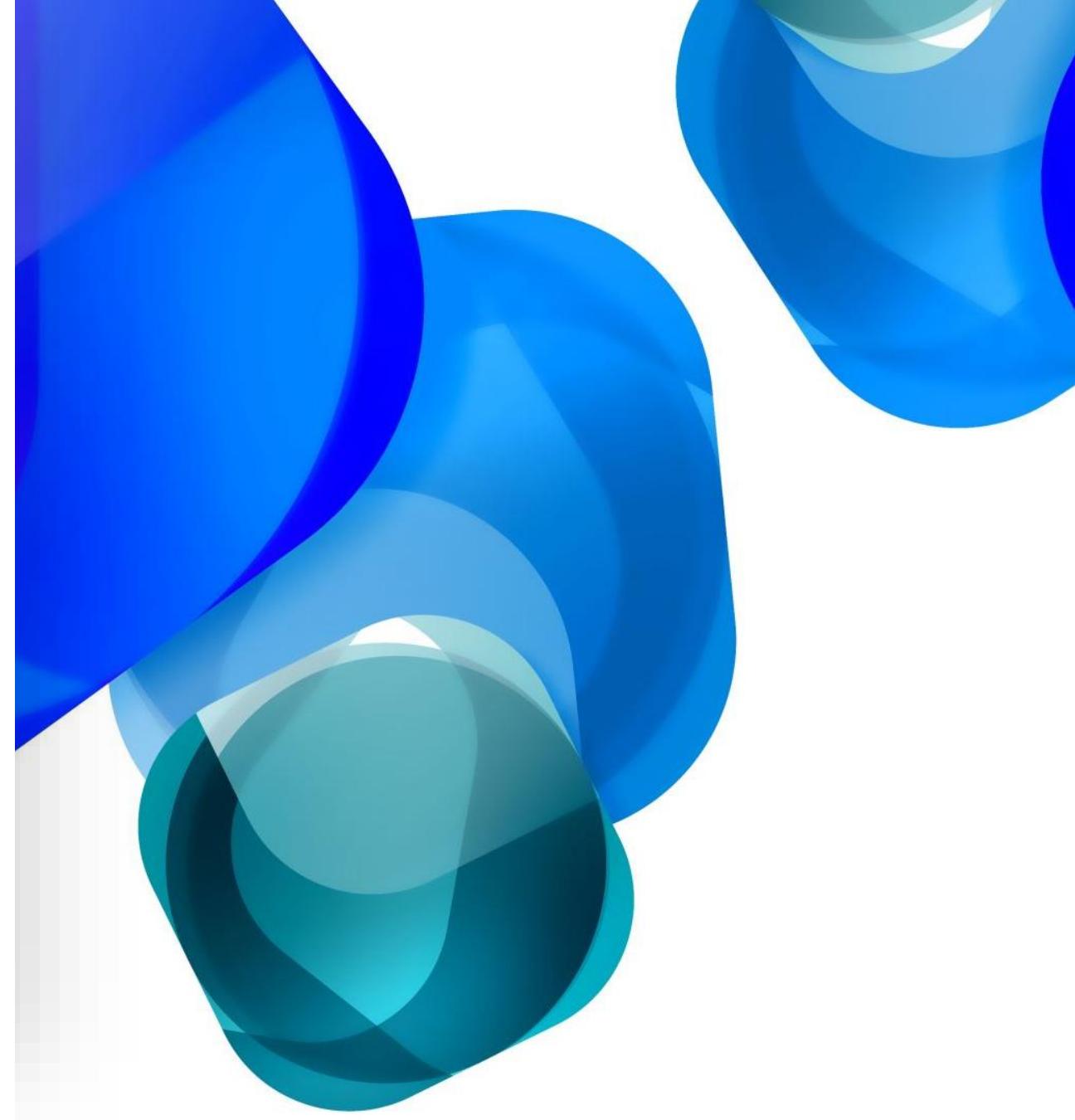
Azure OpenAI - RAG



Demo RAG – with Azure AI search and embeddings

Azure OpenAI on your own data with RAG

- Demo/Lab:
 - [Using your data with Azure OpenAI Service - Azure OpenAI | Microsoft Learn](#)
 - Use your own images: [Use your image data with Azure OpenAI Service in Azure OpenAI studio - Azure OpenAI | Microsoft Learn](#)





Rag - grafikk

[Rag-nice-graphics.docx](#)



Azure AI Studio



Build and
train your
own models



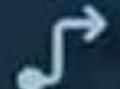
Ground Azure
OpenAI Service
and OSS models
using your data



Built-in
vector
indexing



Retrieval
augmented
generation
made easy



Create
prompt
workflows



AI safety
built-in

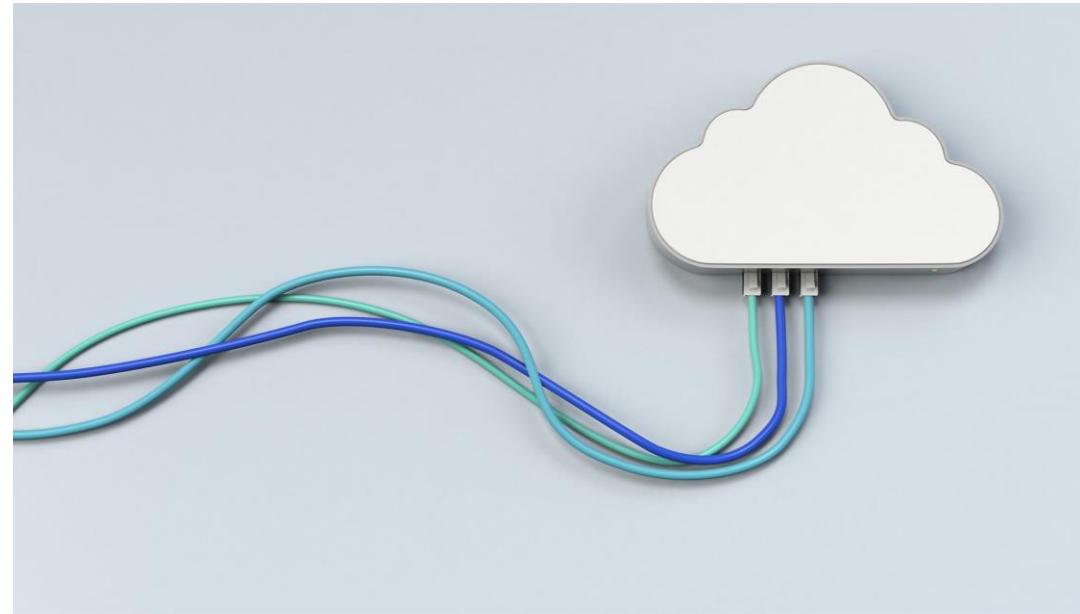
Azure AI studio

Azure AI Studio - demoer



INTRODUCTION TO AZURE PROMPTFLOW

- Azure PromptFlow is an orchestration tool developed by Microsoft for Azure AI Studio.
- It allows developers to coordinate the individual parts of their AI solutions and make them work seamlessly together.
- PromptFlow supports the use of both graphical and code-based development approaches, giving developers the flexibility to choose which approach works best for them.



GENERATIVE AI - LIMITATIONS

Signature



SOME LIMITATIONS OF USING CHATGPT



- Context
- Knowledge
- Manipulation
- Unreliability
- Understanding
- Legal and ethical issues (Responsible AI)

COMMON SENSE AND TACIT KNOWLEDGE

- Large language models, such as ChatGPT, have limited knowledge of the world. This applies to both GPT-3.5 and GPT-4, for example.
- Copilot in Bing, for example, therefore, uses both ChatGPT and Bing Search
- Generative AI models – such as ChatGPT – are just a probability machine.

THE CONVERSATION SITUATION

- Lack of emotional intelligence: While ChatGPT can generate responses that seem empathetic, it does not have true emotional intelligence. It can't detect subtle emotional cues or respond appropriately to complex emotional situations. When people communicate, we use far more than rational language - including body language.
- Limitations in understanding context: ChatGPT has difficulty understanding context, especially sarcasm and humor. While ChatGPT is proficient in language processing, it can struggle to understand the subtle nuances of human communication. For example, if a user were to use sarcasm or humor in their message, ChatGPT may fail to capture the intended meaning and instead provide a response that is inappropriate or irrelevant.



BIAS- WITH CHATGPT AS AN EXAMPLE

- ChatGPT was trained on texts by people all over the world, past and present. Unfortunately, this means that the same real-world biases can also appear in the model.
- ChatGPT has been shown to produce responses that discriminate against gender, race, and minority groups, which the company is trying to curb.
- One way to explain this problem is to point to the data as the problem, blaming humanity for the biases embedded in the internet and beyond. But part of the responsibility also lies with OpenAI and Microsoft, whose researchers and developers choose the data used to train ChatGPT.

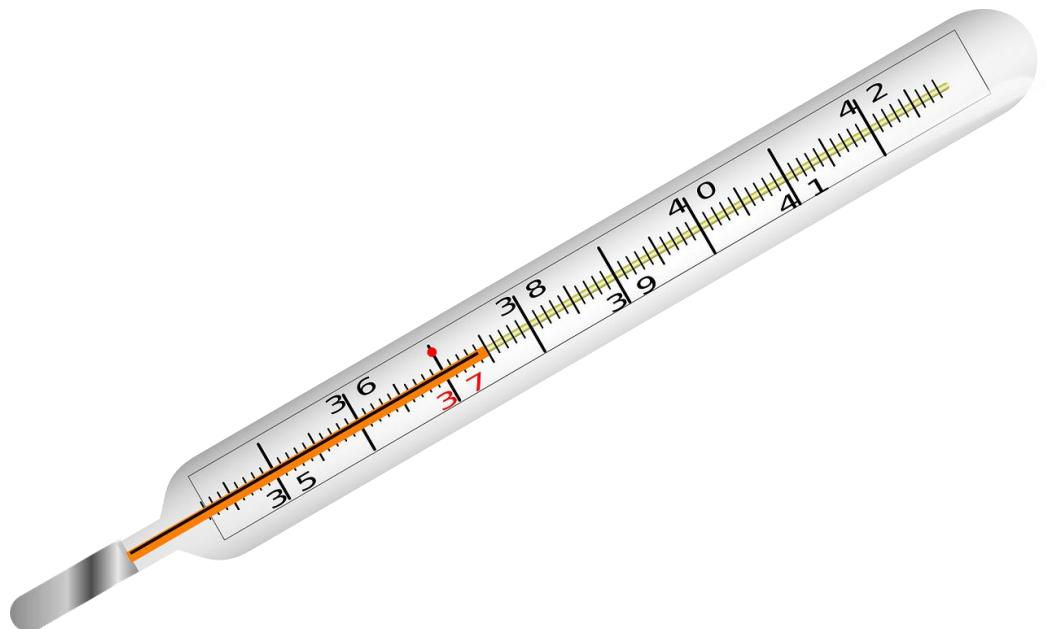


CHATGPT AND HALLUCINATION

- When an AI model "hallucinates," it generates fabricated information in response to a user's questions but presents it as if it is factual and correct.
- Suppose you asked an AI chatbot to write an article about the Statue of Liberty. The chatbot would be hallucinating if it said that the monument was located in California instead of saying that it is in New York.
- But the mistakes are not always so obvious. In response to the Statue of Liberty message, the AI chatbot can also create names of designers who worked on the project or state that it was built in the wrong year.
- This happens because large language models, often referred to as AI chatbots, are trained on massive amounts of data, which is how they learn to recognize patterns and connections between words and topics. They use this knowledge to interpret questions and generate new content, such as text or images.
- However, since AI chatbots are essentially predicting the word that is most likely to come next in a sentence, they can sometimes generate outputs that sound correct but aren't actually true.



TEMPERATURE



EVALUATING AI MODELS IN AZURE AI STUDIO

- Evaluation assesses the performance of generative AI models
- Built-in metrics evaluate single or multi-turn conversations
- Evaluation can be applied to various task types, such as question answering



AZURE AI STUDIO EVALUATION PAGE

- Hub for visualizing, assessing, and optimizing AI model results
- Compare results across multiple evaluation runs
- Identify trends and make data-driven decisions for performance enhancement



MANUAL EVALUATION TOOLS IN AZURE AI STUDIO

- Manual evaluation tools iteratively test and refine prompts
- Test against test data within a single interface
- Critical component for ensuring AI model quality and relevance



MANUAL EVALUATION

Build / contoso-hiking-chatbot / Evaluation / Manual evaluation

Assistant setup

Prompt

You are an AI assistant helping users with queries related to outdoor/camping gear and clothing. Use the following pieces of context to answer the questions about outdoor/camping gear and clothing as completely, correctly, and concisely as possible.

If the question is not related to outdoor/camping gear and clothing, just say Sorry, I only can answer question related to outdoor/camping gear and clothing. So how can I help? Don't try to make up an answer.

If the question is related to outdoor/camping gear and clothing but vague ask for clarifying questions. Do not add documentation reference in the response.

Parameters Add your data

Model gpt-35-turbo-16k

Max response 800

Temperature 0.7

Manual evaluation result

Run Import test data Export Metric evaluation Save results Columns Imported dataset: evaluation_dataset_jsonl_2023-11-15_001008_UTC

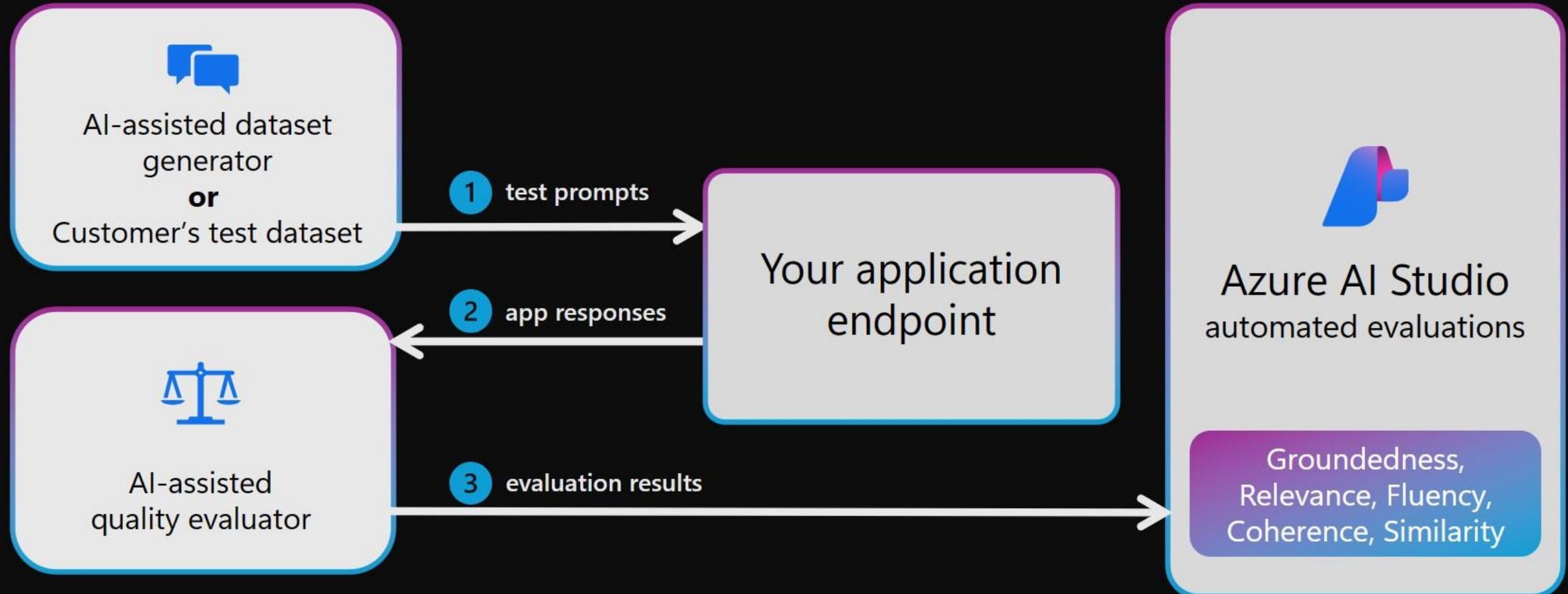
Data rated	Thumbs up	Thumbs down
92.31% (12/13)	84.62% (11/13)	7.69% (1/13)

Input Expected response **Output**

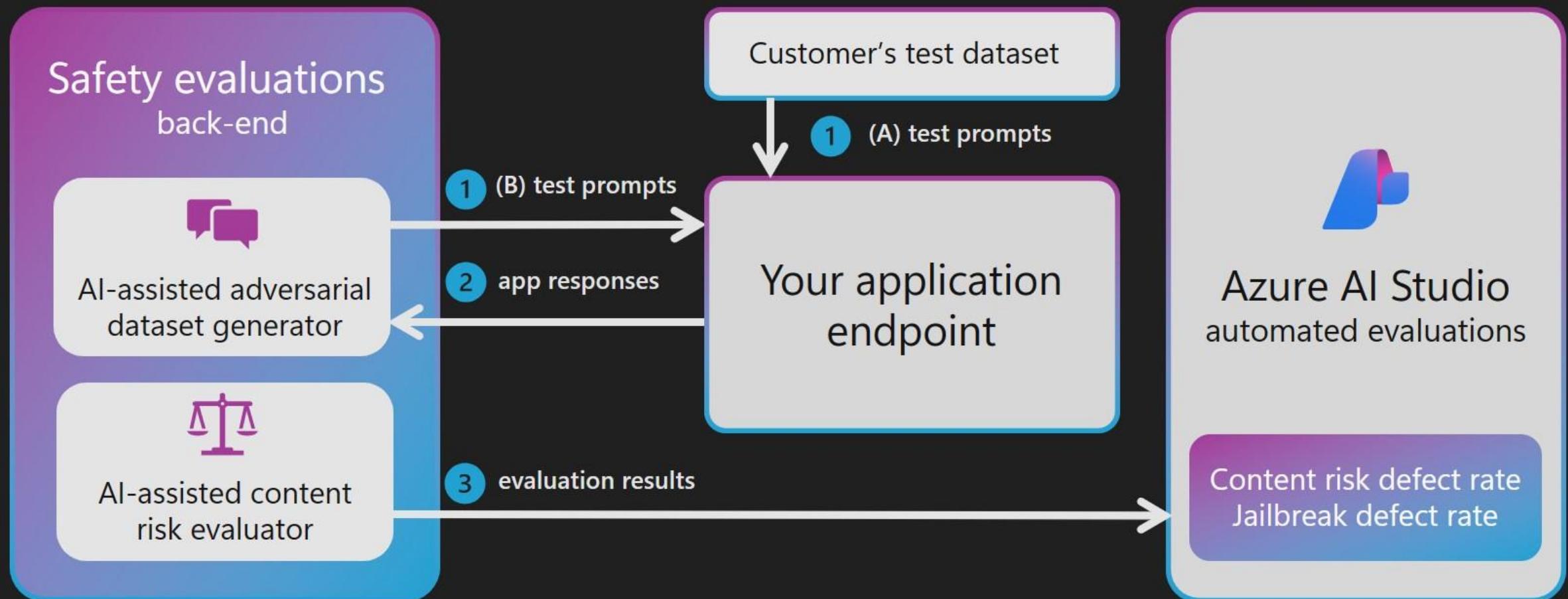
Which tent is the most waterproof?
The Alpine Explorer Tent has the highest rainfly waterproof rating at 3000m
The most waterproof tent among the retrieved documents is the Alpine Explorer Tent, with a rainfly waterproof rating of 3000mm[doc4].
🔗

Which camping table holds the most weight?
The Adventure Dining Table has a higher weight capacity than all of
The weight capacity of a camping table can vary depending on the
🔗

Evaluate generative AI **application quality** with Azure AI Studio

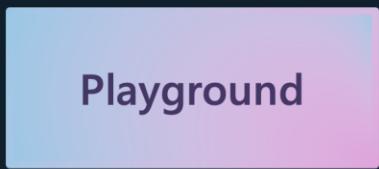


Evaluate generative AI **application safety** with Azure AI Studio

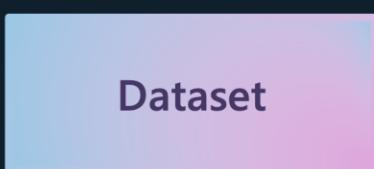


Sample Data (Test Data)

Path 1



Path 3



Playground

Dataset

Manual Evaluation

Manual Evaluation

Automatic Evaluation

Evaluate

Automatic Evaluation

Path 2

Flows

Evaluation Tab

Experiment Tracker

Evaluation UI

Monitoring Tab

- Performance/token Statistics
- Generation Quality and Safety

Deploy

Azure AI Content Safety



Production Data

CONTENT FILTERING

Azure AI Studio Preview

Home Explore Build

contoso-store

- Configure filters
- Additional filter (Optional)
- Review and save

Create a content filter

Configure the threshold levels for your filter

The default content filtering configuration is set to filter at the medium severity threshold for all four content harms categories for both, prompts and completions. [Learn more about Azure AI Content Safety](#)

Give your configuration a custom name: *

CustomContentFilter205

User prompts (Input)

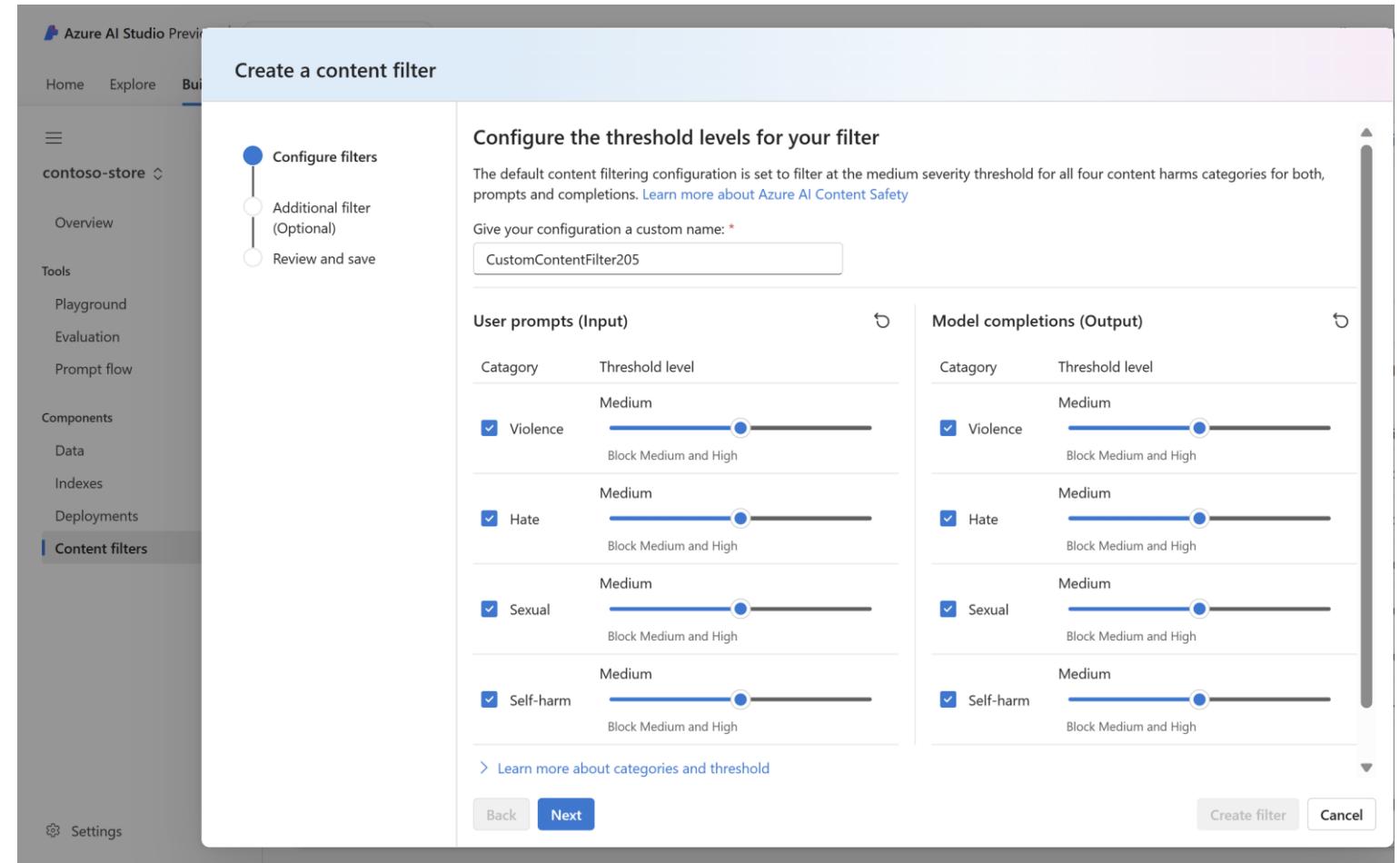
Category	Threshold level
Violence	Medium Block Medium and High
Hate	Medium Block Medium and High
Sexual	Medium Block Medium and High
Self-harm	Medium Block Medium and High

Model completions (Output)

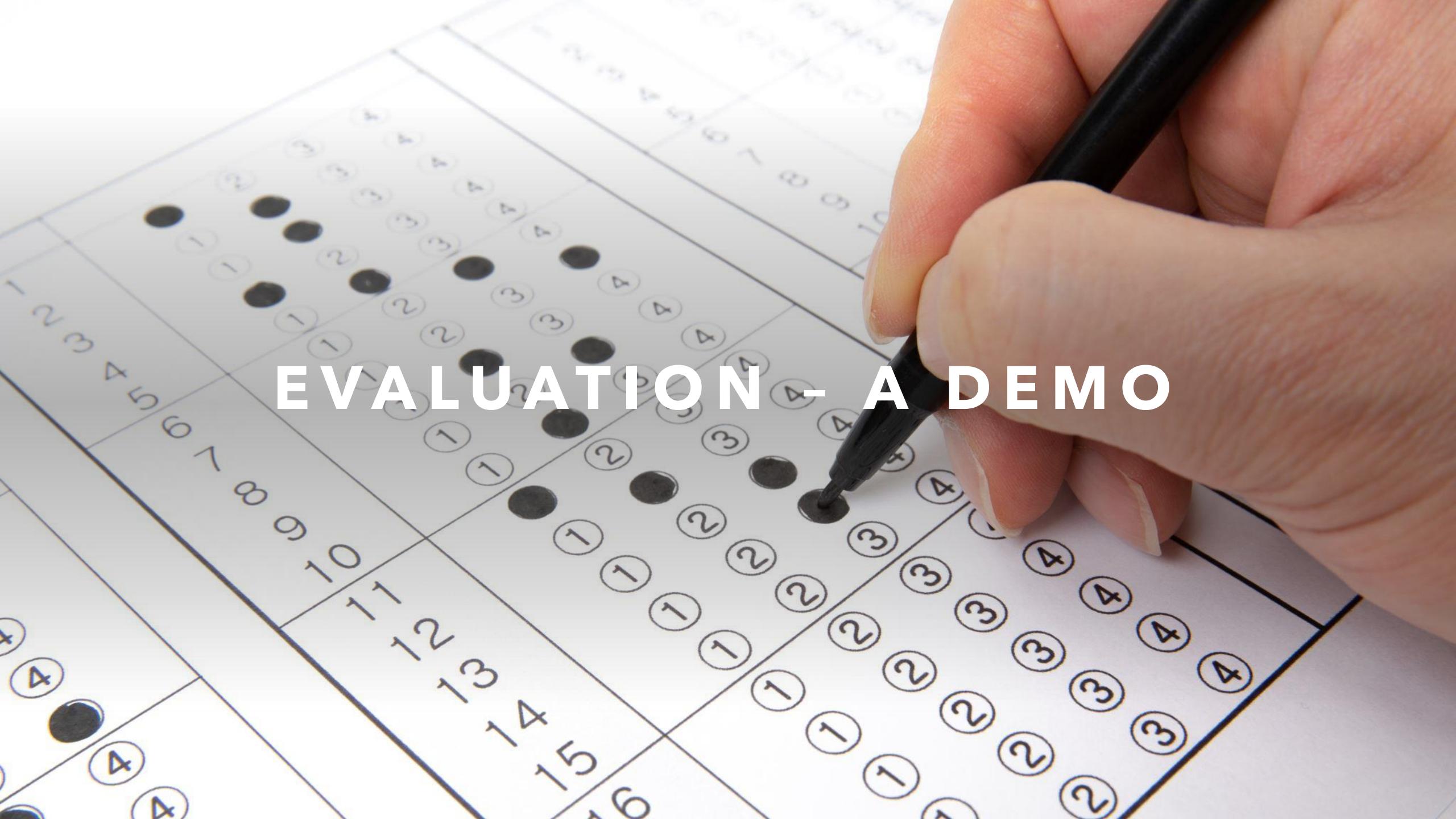
Category	Threshold level
Violence	Medium Block Medium and High
Hate	Medium Block Medium and High
Sexual	Medium Block Medium and High
Self-harm	Medium Block Medium and High

[Learn more about categories and threshold](#)

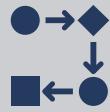
Back Next Create filter Cancel



EVALUATION - A DEMO



Azure AI Studio – se eget oppgaveark



<https://learn.microsoft.com/en-us/azure/ai-studio/tutorials/deploy-chat-web-app>
(stopp når du kommer til Deploy – vi har nok ikke nok ressurser) – og bare les gjennom resten av oppgaven



<https://learn.microsoft.com/en-us/azure/ai-studio/tutorials/deploy-copilot-ai-studio>

Frameworks for LLMs



Demo - Langchain



- Start with a simple notebook: (in folder /kode)
- Start with this: https://python.langchain.com/docs/get_started/quickstart
- Start with this code sample from langchain:
https://python.langchain.com/docs/expression_language/get_started
- Add some more samples from Langchain cookbook:
https://python.langchain.com/docs/expression_language/cookbook/

More code with Azure OpenAI



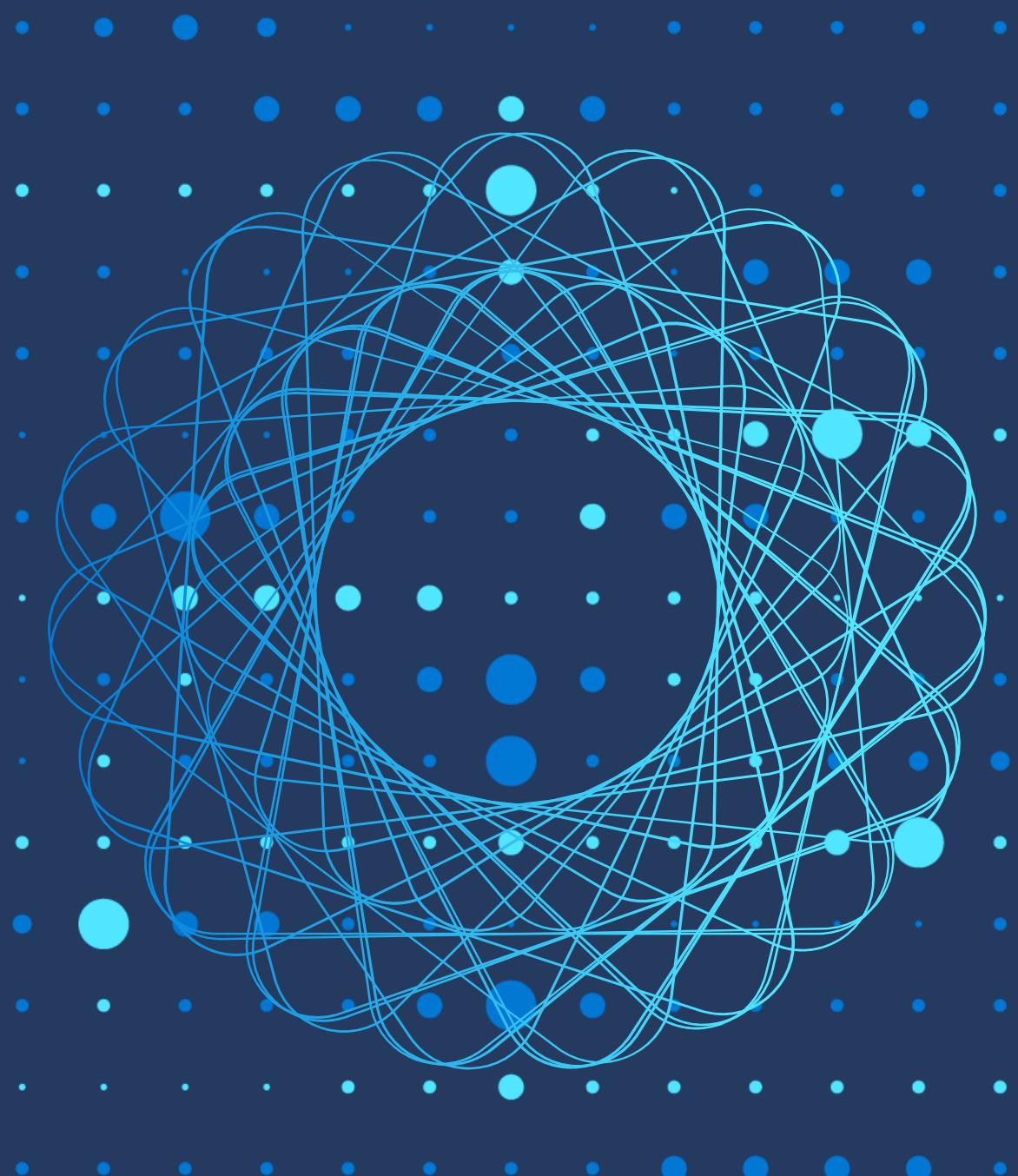


Github – code – let's try out some samples

- https://github.com/Azure/azure-openai-samples/tree/main/quick_start
- [Just an overview:](#)
 - [azure-openai-samples/use_cases/archive/movie_reviews/README.md at main · Azure/azure-openai-samples · GitHub](https://github.com/Azure/azure-openai-samples/blob/main/use_cases/archive/movie_reviews/README.md)

Compute og ressurser

Azure ML Studio



Azure Open AI – flere oppgaver

<https://github.com/Azure-Samples/openai>

1. Last ned og unzip-filene
2. Chat:
 1. Start med å legge inn requirements, nøkkel og endepunkt: <https://github.com/Azure-Samples/openai/blob/main/Basic%20Samples/Chat/README.md>
 2. Åpne og test fortløpende de enkelte notebooks:
 1. Basic_chatcompletions...
 2. ChatGPT_managing...
 3. Chat_with_your_own_data..

Promptflow – tutorials

<https://github.com/microsoft/promptflow>

Semantic Kernel: Just a demo: Create your own local copilot

<https://learn.microsoft.com/en-us/semantic-kernel/chat-copilot/getting-started?tabs=Windows%2CPowershell>



Labb/Oppgaver

- <https://github.com/MicrosoftLearning/mslearn-ai-services>
- <https://github.com/MicrosoftLearning/mslearn-ai-vision>
- <https://github.com/MicrosoftLearning/mslearn-ai-language>
- <https://github.com/MicrosoftLearning/mslearn-ai-document-intelligence>
- <https://github.com/MicrosoftLearning/mslearn-knowledge-mining/>
- <https://github.com/MicrosoftLearning/mslearn-openai>