

Azure AI Foundry

Lab

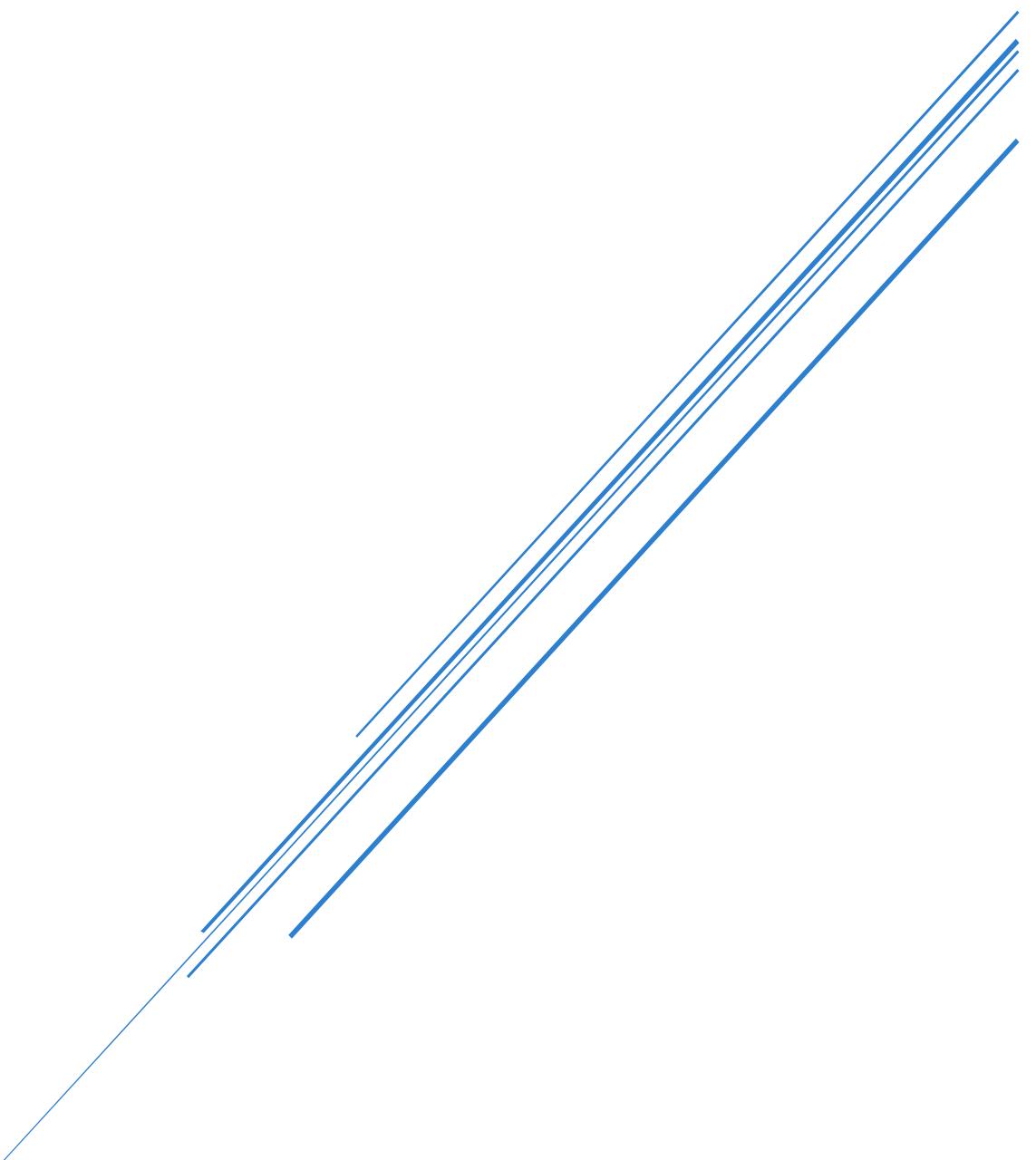


Table des matières

Environment setup.....	2
Create a new Project	2
Action	3
Deploy a model.....	8
Action	8
Retrieval Augmented Generation (RAG).....	15
What is RAG?	15
How does RAG work?	15
What is an index and why do I need it?.....	15
Action	16
Create a conversational RAG flow	24
Action	25
Evaluation	33
What is an evaluation?.....	33
Action - Evaluate your Chat flow.....	33
Discover Content Safety.....	44
Responsible AI Principles	44
Content Filtering System.....	45
Mitigation Layers	45
Action	47

Environment setup

To deploy this lab, an **Azure subscription is required**, where you can create an AI Project along with its AI Hub Resource, a Content Safety service, and an AI Search service.

Lab Steps

1. Use Azure AI Foundry Playground.
2. Work with an Open Source LLM Model.
3. RAG – connect your data.
4. Create a Prompt Flow flow
5. Evaluate your model
6. Test the prompt in Content Safety.

Create an AI Project and AI Hub Resources

Let's start by creating a project in Azure AI Foundry.

Go to your browser and type: <https://ai.azure.com>.

Logging with your Microsoft account.

The screenshot shows the Azure AI Foundry interface. On the left, there's a sidebar with sections like 'Jump into a project in Azure AI Foundry' (listing two projects: 'project_rag_lab2' and 'project-labastudio'), 'Work outside of a project' (with cards for 'Focused on Azure OpenAI Service?' and 'Chat playground'), and 'Find it fast' (with cards for 'Quota management', 'Model catalog and benchmarks', 'Safety and security', and 'Content Understanding'). On the right, there's a 'Help' sidebar with sections like 'Watch a tutorial', 'Overview', 'What are AI services?', 'Azure AI Studio architecture', 'Quick starts', and 'Tutorials'. A 'Launch Get Started tutorial' button is also present.

Create a new Project

Definition

A **project** is an organizational container that has tools for AI customization and orchestration. It lets you organize your work, save state across different tools like prompt

flow, and collaborate with others. For example, you can share uploaded files and connections to data sources.

Multiple projects can use a hub, and multiple users can use a project. A project also helps you keep track of billing and manage access and provides data isolation. Every project uses dedicated storage containers to let you upload files and share it with only other project members when using the 'data' experiences.

Action

- **Click on:** <https://ai.azure.com/>
- **Click on:** “+ Create project”

View all projects + Create project ? Help

Create a project

Projects are easy-to-manage containers for your work—and the key to collaboration, organization, and connecting data and other services.

Project name * ⓘ
frgail-1489

Hub ⓘ
hub_lab2 [Create new hub](#)

[Create](#) [Cancel](#)

- Type a name for your project.
- Click on: “**Create new hub**”

 **Definition: Hubs** are the primary top-level Azure resource for AI Studio and provide a central way for a team to govern security, connectivity, and computing resources across playgrounds and projects. Once a hub is created, developers can create projects from it and access shared company resources without needing an IT administrator's repeated help.

You can create and manage a hub from the Azure portal or from the AI Foundry.



If you want to **create a secure Hub**, you must create the Hub from the portal:

[How to create and manage an Azure AI Foundry hub - Azure AI Foundry | Microsoft Learn](#)

- Type a name for your hub

A hub is the collaboration environment for your team to share your project work, model endpoints, compute, connections, and security settings.

Name

Hub-AI

[Learn more](#)

[Next](#)

[Cancel](#)

- Click on **Customize**

Create a project

Projects are easy-to-manage containers for your work—and the key to collaboration, organization, and connecting data and other services.

Project name * ⓘ

frgail-7494

Hub ⓘ

[Create new hub](#)

Select or search by name

▼

✓ Azure resources to be created (new: Hub-AI + 4)

[Customize](#)

Subscription

MCAPS-Hybrid-REQ-40894-2022-frgail

Hub

(new) Hub-AI

Resource group

(new) rg-frgail-1297_ai

Data storage

(new) Hub, Storage, Key Vault, AI Services

Location

eastus

Public network access

Enabled

[View resource and pricing details](#)

ⓘ Do you need to customize the security or storage resources? [Go to Azure Portal](#)

[Create](#)

[Cancel](#)

- Fill the cells

Create a project

1 Project details
2 Create a hub
3 Review and finish

Create a hub for your projects
A hub is the collaboration environment for your team to share your project work, model endpoints, compute, connections, and security settings. [Learn more](#)

Do you need to customize security or the [dependent resources](#) of your hub? [Go to Azure Portal](#)

Hub name *
Hub-AI

Subscription * ⓘ [Create new subscription](#)
MCAPS-Hybrid-REQ-40894-2022-frgail

Resource group * ⓘ [Create new resource group](#)
labaistudio

Location * ⓘ [Help me choose](#)
East US

Models availability by region

Connect Azure AI Services or Azure OpenAI * ⓘ [Create new AI Services](#)
(new) ai-labaifoundry

Connect Azure AI Search ⓘ [Create new AI Search](#)
(new) aisearch-labaifoundry

[Back](#) [Next](#) [Create](#) [Cancel](#)

- Click on **Create**

Create a project

1 Project details
2 Create a hub
3 Review and finish

Review and finish

The following resources will be created for you, along with required dependencies. The creation of the first hub and project may take a few minutes to complete. [Learn more about hubs and dependencies](#).

Hub
Name: Hub-AI
Subscription: MCAPS-Hybrid-REQ-40894-2022-frgail
Resource group: labaistudio
Location: eastus

Project
Name: frgail-7494
Subscription: MCAPS-Hybrid-REQ-40894-2022-frgail
Resource group: labaistudio

AI Services
Name: ai-labaifoundry

AI Search
Name: aisearch-labaifoundry

[Back](#) [Create](#) [Cancel](#)

- Click on **Open in management center**

The screenshot shows the Azure AI Foundry interface. On the left, there's a sidebar with various project management options like Model catalog, Playgrounds, AI Services, and Code. The main area displays the project name 'frgail-7494' and its endpoints and keys. On the right, under 'Project details', there's a section for managing project settings, including 'Add users', 'Connect resources', 'View quota', and 'Track costs'. A prominent red box highlights the 'Open in management center' button.

 **Definition:** The **management center** is a centralized control panel within the Azure AI Foundry portal. It streamlines governance and management activities, providing a simplified and centralized experience for cross-functional teams. From the management center, you can manage various aspects such as projects, resources, quotas, usage metrics, access rights, and permissions.

The screenshot shows the Azure Management Center interface for the same project. The left sidebar includes 'Management center' and 'Project (frgail-7494)' sections. The main area shows the project overview with sections for 'Models + endpoints', 'Project users', and 'Connected resources'. A large red box highlights the 'Connected resources' section, which lists various Azure services connected to the project. An orange arrow points from this section down to the 'Total cost' section at the bottom, which provides information about consumed compute and other resources.

Name	Type	Target	Key	Authentication type
AzureAISeach	Azure AI Search (Cogniti...	https://aisearch-labaifoundry721835953777.se...	API key
ai-labaifoundry721835953777_aoui	Azure OpenAI	https://ai-labaifoundry721835953777.openai.a...	API key
ai-labaifoundry721835953777	AI Services	https://ai-labaifoundry721835953777.cognitive...	API key
frgail-7494/workspaceblobstore	Azure Blob Storage	https://sthubai721835953777.blob.core.windo...	--	SAS
frgail-7494/workspaceartifactstore	Azure Blob Storage	https://sthubai721835953777.blob.core.windo...	--	SAS

4 connected resources have been created:

- **Azure AI Search** to build an index and store vectors.
- **AI Services** to access AI Services (Speech, Language+Translator, Vision, Content Safety)
- **Azure OpenAI service** to access Azure OpenAI models
- **Azure Blob Storage** to store data and artifacts.
 - **Workspace Artifact Store:**
 - Primarily used to store various artifacts related to your AI projects, such as datasets, models, logs, and other files.
 - Each project has its own dedicated storage containers within the workspace artifact store, maintaining data isolation and security.
 - **Workspace Blob Store :**
 - Acts as the default blob storage for the workspace, used for general data storage needs.
 - Typically used for storing large amounts of unstructured data, such as text, images, and binary data.

In summary, the Hub's key points:

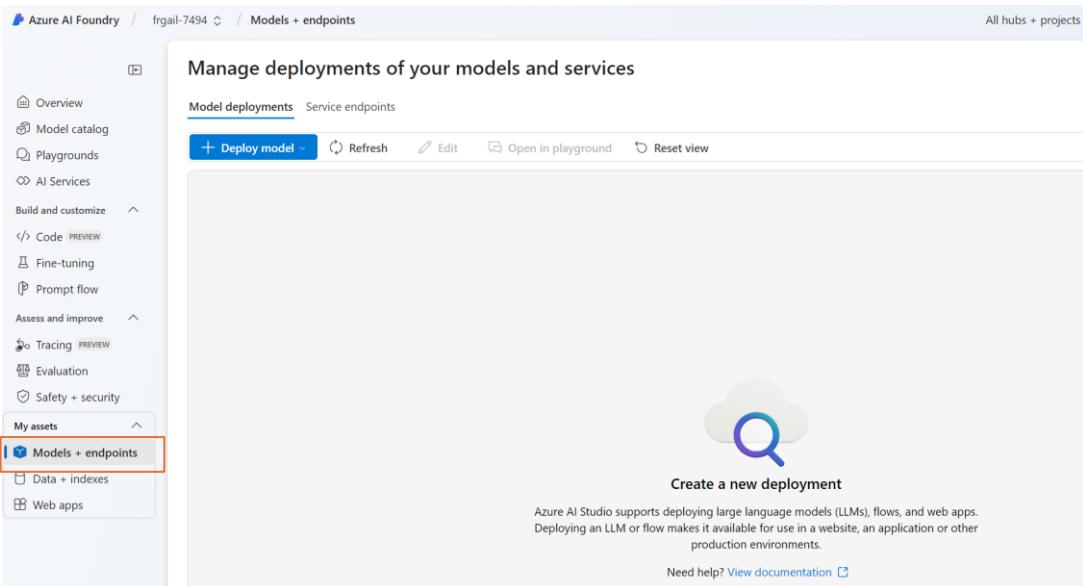
1. **Centralized Management:** A hub provides a unified environment where teams can manage security, connectivity, and computing resources across various AI projects.
2. **Collaboration:** It allows multiple developers and data scientists to collaborate on machine learning projects, sharing resources and configurations easily.
3. **Resource Sharing:** Hubs enable the sharing of Azure resources like storage accounts, model endpoints, and more, without needing repeated IT intervention.
4. **Security and Compliance:** You can set up and enforce security policies, network configurations, and compliance requirements at the hub level, which are then inherited by all projects under the hub.
5. **Project Organization:** Projects created within a hub can be customized and isolated, allowing for organized workspaces that help in managing data, access, and billing.
6. **Ease of Use:** Hubs simplify the process of setting up environments for AI development, making it easier to prototype, build, and deploy AI applications.

Deploy a model

After creating your AI Project, the first step is to create a deployment of an Azure OpenAI model so you can start experimenting with the prompts you will use in your application.

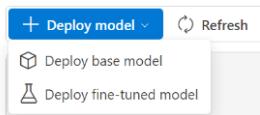
Action

- In the project you just created, click on **Models + endpoints**:



The screenshot shows the Azure AI Foundry interface. The left sidebar has sections like Overview, Model catalog, Playgrounds, AI Services, Build and customize, Assess and improve, and Safety + security. Under My assets, 'Models + endpoints' is highlighted with a red box. The main area is titled 'Manage deployments of your models and services' and shows the 'Model deployments' tab selected. It features a 'Deploy model' button, a search bar, and a 'Create a new deployment' section with a cloud and magnifying glass icon. Below it, there's a note about Azure AI Studio supporting large language models (LLMs), flows, and web apps.

- Select Deploy base model



The screenshot shows a dropdown menu with two options: 'Deploy base model' and 'Deploy fine-tuned model'. The 'Deploy base model' option is selected.

Select a model

Choose a model to create a new deployment. For flows and other resources, create a deployment from their respective list. [Go to model catalog.](#)

Models: 1812 | [Collections](#) | [Deployment options](#) | [Inference tasks](#) | [Show description](#)

The screenshot shows a list of models on the left side of the interface. Each model entry includes a small icon, the model name, its purpose, and two circular buttons. A search bar is at the top. On the right, there's a large purple circular icon with a blue cube inside, and the text "Select a model to see description". At the bottom right are "Confirm" and "Cancel" buttons.

Model	Purpose	Action Buttons
gpt-4o-realtime-preview	Audio generation	○ ○
gpt-4	Chat completion	○ ○
gpt-35-turbo	Chat completion	○ ○
o1-preview	Chat completion	○ ○
o1-mini	Chat completion	○ ○
gpt-4o-mini	Chat completion	○ ○
gpt-4o	Chat completion	○ ○



In Azure AI Studio, there are **two primary types of model deployment:**

This screenshot shows the "Deployment options" section of the interface. It includes a title, a description, and two checkbox options: "Managed compute" and "Serverless API".

1. Serverless API (MaaS : Model as a Service)

- **Description:** This deployment type allows you to deploy your model as a service without managing the underlying infrastructure.
- **Billing:** You are billed based on the number of tokens processed (pay-as-you-go).
- **Scalability:** Automatically scales to handle varying loads.
- **Use Case:** Ideal for applications where you need a flexible, cost-effective solution that can handle unpredictable traffic.

2. Managed Compute (MaaP : Model as a Platform)

- **Description:** This deployment type involves hosting your model on dedicated virtual machines within your Azure subscription.
- **Billing:** You are billed for the virtual machine core hours used.
- **Control:** Provides more control over the infrastructure, including the ability to configure the number of instances and manage capacity.

- **Use Case:** Suitable for scenarios requiring consistent performance, higher control over the environment, and potentially lower costs for predictable workloads.
- In the **search bar**, type **Mistral Large 2407** and click on **Confirm**:

Select a model

Choose a model to create a new deployment. For flows and other resources, create a deployment from their respective list. [Go to model catalog](#)

Models: 263 Collections Deployment options Inference tasks Show description

Mistral large 2407

Mistral-large-2407

Mistral-large-2407 Chat completion

Mistral-large Chat completion

mistral-community-mistral... Text generation

Mistral-small Chat completion

Mistral-Nemo Chat completion

mistralai-Mistral-7B-v01 Text generation

teknum-openhermes-2.5... Text generation

Task: Chat completion

Mistral Large (2407) is an advanced Large Language Model (LLM) with state-of-the-art reasoning, knowledge and coding capabilities.

Multi-lingual by design. Dozens of languages supported, including English, French, German, Spanish, Italian, Chinese, Japanese, Korean, Portuguese, Dutch and Polish

Proficient in coding. Trained on 80+ coding languages such as Python, Java, C, C++, JavaScript, and Bash. Also trained on more specific languages such as Swift and Fortran

Agent-centric. Best-in-class agentic capabilities with native function calling and JSON outputting

Advanced Reasoning. State-of-the-art mathematical and reasoning capabilities

Prev Next

Confirm Cancel

Serverless API deployment for Mistral-large-2407

Overview Pricing and terms

 Mistral Large 2407 is offered by Mistral AI through the Azure Marketplace. View the pricing and terms tab to learn about pricing and terms of use. [Learn more about Models as a Service](#)

Current Project resource
axway-genai

I acknowledge that the [Microsoft Purchase Policy](#) applies to my use of this model and I agree to use only dummy or artificial data sets in a non-production system solely to demo, test and/or evaluate the model. I understand no live or production data may be Processed by an AI system at any time without enrollment and compliance as a Supplier with all applicable Microsoft Policies.

Terms of use

By clicking "Subscribe and Deploy", I (a) agree to the legal terms and privacy statements associated with each Marketplace offering above, (b) authorize Microsoft to charge or bill my current payment method for the fees associated with my use of the offerings, including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offerings, (c) agree that Microsoft may share [my contact information and transaction details \(including usage volume associated with my Azure subscription\)](#)

Azure Marketplace Terms

Subscribe and Deploy Cancel

- Check the cell:

I acknowledge that the [Microsoft Purchase Policy](#) applies to my use of this model and I agree to use only dummy or artificial data sets in a non-production system solely to demo, test and/or evaluate the model. I understand no live or production data may be Processed by an AI system at any time without enrollment and compliance as a Supplier with all applicable Microsoft Policies.

- Click on tab “**Pricing and terms**” to check the pricing:

Serverless API deployment for Mistral-large-2407

Overview [Pricing and terms](#)



Mistral Large 2407 is an advanced Large Language Model (LLM). Start using Mistral Large 2407 today on Azure AI Studio and Azure Machine Learning for any language-based application that requires reasoning, understanding, coding and text-generation capabilities.
[Read more...](#)

Pricing

paygo-inference-output-tokens:
\$0 per 1,000 tokens

paygo-inference-input-tokens:
\$0 per 1,000 tokens

Legal

[Privacy policy](#) [License agreement](#)

[Subscribe and Deploy](#)

Cancel

- Click on “**Subscribe and Deploy**”

- Give a name to your deployment:

Deploy Mistral-large-2407

Deployment name *

Mistral-large-2407-labai

[Deploy](#)

Cancel

Mistral-large-2407-labai

Deployment info

- Name: Mistral-large-2407-labai
- Provisioning state: Succeeded
- Created by: franck.gaillard@microsoft.com
- Last updated on: Dec 4, 2024 4:05 PM
- Model: Mistral-large-2407

Endpoint

- Target URI: https://Mistral-large-2407-labai.eastus.models.ai.azure.com
- Key: [REDACTED]
- Compute type: Consumption
- Swagger URI: https://Mistral-large-2407-labai.eastus.models.ai.azure.com/swagger.json

Useful links for application development

- Code sample repository: [Code sample repository](#)
- Tutorial: [Tutorial](#)

API Routes

- Azure AI model inference: Chat Completion
https://Mistral-large-2407-labai.eastus.models.ai.azure.com/chat/completions
- Mistral: Chat Completion
https://Mistral-large-2407-labai.eastus.models.ai.azure.com/v1/chat/completions

Monitoring & safety

- Azure AI Content Safety: Enabled

- Click on **Open in playground**:

Chat playground

Setup

Deployment *: Mistral-large-2407-labai

Give the model instructions and context: You are an AI assistant that helps people find information.

Chat session

Start typing here

- We will run an example where the model will help us **summarize and extract information from a conversation** between a customer and a representative of a telco company.

Copy the following prompt into the system message field of the playground:

You're an AI assistant that helps telco company to extract valuable information from their conversations by creating JSON files for each conversation transcription you receive. You always try to extract and format as a JSON:

1. Customer Name [name]
2. Customer Contact Phone [phone]

3. Main Topic of the Conversation [topic]
4. Customer Sentiment (Neutral, Positive, Negative)[sentiment]
5. How the Agent Handled the Conversation [agent_behavior]
6. What was the FINAL Outcome of the Conversation [outcome]
7. A really brief Summary of the Conversation [summary]

Only extract information that you're sure. If you're unsure, write "Unknown/Not Found" in the JSON file.

After copying, select “**Apply changes**”.

The screenshot shows the Azure AI Foundry interface for a 'Chat playground'. The left sidebar has a tree view with 'Playgrounds' selected. The main area has a 'Setup' section with a 'Deployment' dropdown set to 'Mistral-large-2407-labai'. Below it is a 'Give the model instructions and context' section containing a JSON template:

```

    You're an AI assistant that helps telco company to
    extract valuable information from their conversations by
    creating JSON files for each conversation transcription
    you receive. You always try to extract and format as a
    JSON:
    1. Customer Name [name]
  
```

An 'Apply changes' button is highlighted with a red box. To the right is a 'Chat session' window with a message input field at the bottom.

Then type the following text in the chat session and click the send button:

Agent: Hello, welcome to Telco's customer service. My name is Juan, how can I assist you?

Client: Hello, Juan. I'm calling because I'm having issues with my mobile data plan. It's very slow and I can't browse the internet or use my apps.

Agent: I'm very sorry for the inconvenience, sir. Could you please tell me your phone number and your full name?

Client: Yes, sure. My number is 011-4567-8910 and my name is Martín Pérez.

Agent: Thank you, Mr. Pérez. I'm going to check your plan and your data usage. One moment, please.

Client: Okay, thank you.

Agent: Mr. Pérez, I've reviewed your plan and I see that you have contracted the basic plan of 2 GB of data per month. Is that correct?

Client: Yes, that's correct.

Agent: Well, I inform you that you have consumed 90% of your data limit and you only have 200 MB available until the end of the month. That's why your browsing speed has been reduced.

Client: What? How is that possible? I barely use the internet on my cell phone. I only check my email and my social networks from time to time. I don't watch videos or download large files.

Agent: I understand, Mr. Pérez. But keep in mind that some applications consume data in the background, without you realizing it. For example, automatic updates, backups, GPS, etc.

Client: Well, but they didn't explain that to me when I contracted the plan. They told me that with 2 GB I would have enough for the whole month. I feel cheated.

Agent: I apologize, Mr. Pérez. It was not our intention to deceive you. I offer you a solution: if you want, you can change your plan to a higher one, with more GB of data and higher speed. This way you can enjoy a better browsing experience.

Client: And how much would that cost me?

Agent: We have a special offer for you. For only 10 pesos more per month, you can access the premium plan of 5 GB of data and 4G speed. Are you interested?

Client: Mmm, I don't know. Isn't there another option? Can't you give me more speed without charging me more?

Agent: I'm sorry, Mr. Pérez. That's the only option we have available. If you don't change your plan, you'll have to wait until next month to recover your normal speed. Or you can buy an additional data package, but it would be more expensive than changing plans.

Client: Well, let me think about it. Can I call later to confirm?

Agent: Of course, Mr. Pérez. You can call whenever you want. The number is the same one you dialed now. Is there anything else I can help you with?

Client: No, that's all. Thank you for your attention.

Agent: Thank you, Mr. Pérez. Have a good day. Goodbye.

The screenshot shows the Azure AI Foundry interface with the 'Playgrounds' section selected. On the left, there's a sidebar with various options like Overview, Model catalog, Playgrounds, AI Services, etc. The main area is titled 'Chat playground' and shows a 'Setup' section for a deployment named 'Mistral-large-2407-labai'. It includes fields for giving model instructions and context, and a 'Parameters' section. To the right is a 'Chat session' window displaying a transcript between a Client and an Agent. The Client asks about their number and name, and the Agent responds with a detailed explanation. Below the transcript is a JSON representation of the extracted information.

```

{
  "name": "Martin Pérez",
  "phone": "011-4567-8910",
  "topic": "Issues with mobile data plan speed",
  "sentiment": "Negative",
  "agent_behavior": "Polite and helpful, provided a solution",
  "outcome": "Customer will consider upgrading the plan",
  "summary": "Customer complained about slow mobile data, agent explained data usage and offer"
}

```

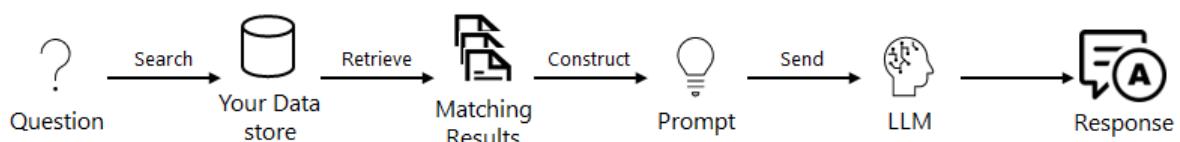
Retrieval Augmented Generation (RAG)

What is RAG?

Large language models (LLMs) like ChatGPT are trained on public internet data that was available at the point in time when they were trained. They can answer questions related to the data they were trained on. This public data might not be sufficient to meet all your needs. You might want questions answered based on your private data. The public data might simply have gotten out of date. The solution to this problem is Retrieval Augmented Generation (RAG), a pattern used in AI that uses an LLM to generate answers with your own data.

How does RAG work?

When a user asks a question, the data store is searched based on user input. The user question is then combined with the matching results and sent to the LLM using a prompt (explicit instructions to an AI or machine learning model) to generate the desired answer. This can be illustrated as follows :



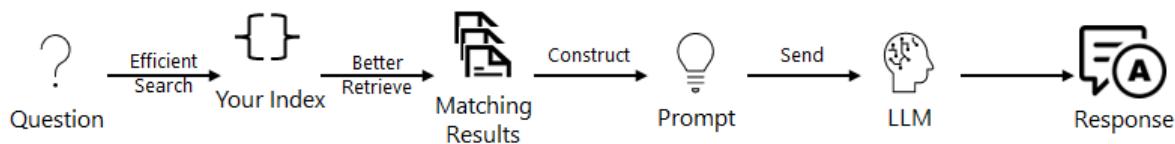
What is an index and why do I need it?

RAG uses your data to generate answers to the user question. For RAG to work well, we need to find a way to search and send your data in an easy and cost-efficient manner to the LLMs. This is achieved by using an index.

An index is a data store that allows you to search data efficiently. This is very useful in RAG. An index can be optimized for LLMs by creating vectors (text data converted to number sequences using an embedding model).

A good index usually has efficient search capabilities like keyword searches, semantic searches, vector searches or a combination of these.

This optimized RAG pattern can be illustrated as follows:



Azure AI provides an index asset to use with RAG pattern. The index asset contains important information like where is your index stored, how to access your index, what are the modes in which your index can be searched, does your index have vectors, what is the embedding model used for vectors etc.

The Azure AI index uses Azure AI Search as the primary and recommended index store. Azure AI Search is an Azure resource that supports information retrieval over your vector and textual data stored in search indexes.

Action

In this lab, you **deploy an enterprise chat web app** that uses your own data with a large language model in AI Studio. Your data source is used to help ground the model with specific data. **Grounding** means that the model uses your data to help it understand the context of your question. You're not changing the deployed model itself. Your data is stored separately and securely in your original data source.

The steps in this lab are:

1. Deploy and test a chat model without your data.
2. Add your data.
3. Test the model with your data.
4. Deploy your web app.

Prerequisites

- An Azure **subscription**.
- An AI Studio **hub**, **project**, and **deployed Azure OpenAI chat model**. Complete the AI Studio playground [quickstart to create these resources](#) if you haven't already.
- An Azure AI Search service connection to index the sample product data.
- You need a local copy of product data. The Azure-Samples/rag-data-openai python-promptflow repository on GitHub contains sample retail product information that's relevant for this tutorial scenario. Specifically, the product_info_11.md file contains product information about the TrailWalker hiking shoes that's relevant for this tutorial example. Download the example Contoso Trek retail product data in a ZIP file to your local machine. [Download the example Contoso Trek retail product data in a ZIP file](#)

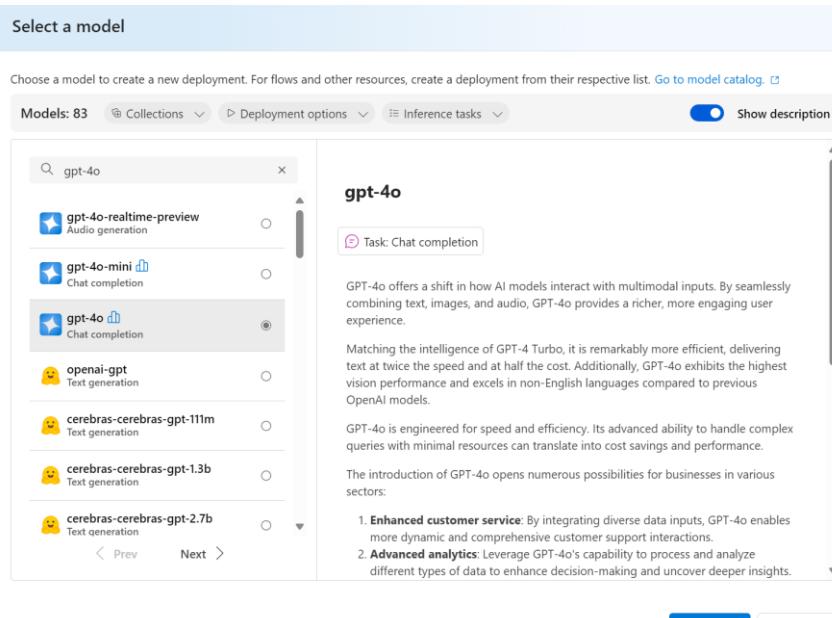
- You need to have Microsoft.Web resource provider registered in the selected subscription, to be able to deploy to a web app. To verify:
 - **Sign in to the Azure Portal:** Go to the Azure Portal and log in with your credentials.
 - **Navigate to Subscriptions:**
 - In the Azure Portal menu, search for **Subscriptions** and select it.
 - **Select Your Subscription:**
 - Choose the subscription where you want to register the resource provider.
 - **Open Resource Providers:**
 - In the left-hand menu, under **Settings**, click on **Resource providers**.
 - **Register Microsoft.Web:**
 - In the list of resource providers, find **Microsoft.Web**.
 - Click on **Register** to register the resource provider.

Add your data and try the chat model

Follow these steps to add your data in the chat playground to help the assistant answer questions about your products. You're not changing the deployed model itself. Your data is stored separately and securely in your Azure subscription.

Deploy a GPT-4o model

1. Open your project
2. Click on Models + endpoints
3. Click on Deploy model / Deploy base model
4. In the **search bar**, type **GPT-4o**, select **gpt-4o** and click on **Confirm**:



5. Click on Deploy:

Deploy model gpt-4o

Deployment name *
gpt-4o-labai

Deployment type
Global Standard

Global Standard: Pay per API call with the highest rate limits. Learn more about [Global deployment types](#).

Data might be processed globally, outside of the resource's Azure geography, but data storage remains in the AI resource's Azure geography. Learn more about [data residency](#).

Deployment details

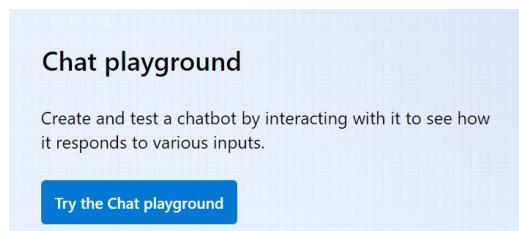
Model version 2024-08-06	Connected AI resource ai-hublab2013149793759_aoai
Project project_rag_lab2	Authentication type Key
Capacity 10K tokens per minute (TPM)	Resource location East US
Content safety DefaultV2	

Customize

Deploy **Cancel**

Deploy a RAG model

1. Select Playgrounds, and click on Try the Chat playground:



2. Select your deployed chat model from the Deployment dropdown.

Azure AI Foundry / project_RAG_lab2 / Playgrounds / Chat playground

All hubs + projects Project project_rag_lab2

← Chat playground

Setup Hide

Deployment * [Create new deployment](#)
gpt-4o (version:2024-05-13)

Give the model instructions and context
You are an AI assistant that helps people find information.

[Apply changes](#) [Generate prompt](#)

Chat history

Start with a sample prompt

Creative storytelling
Write a short story about a time traveler who accidentally changes a major historical event.

Recipe creation
Invent a recipe for a dish that combines flavors from two different cuisines.

Poetry generation
Compose a poem about the beauty of nature in autumn.

Type user query here. (Shift + Enter for new line)

11/128000 tokens to be sent

3. On the left side of the chat playground, select **Add your data > + Add a new data source**.

The screenshot shows the Azure AI Chat playground interface. On the left, there's a sidebar with 'Setup' and 'Deployment' sections. The 'Add your data' section is highlighted with a red border. To the right, there's a main area titled 'Chat history' with a message bubble icon. Below it, a callout box points to the 'Start with a sample prompt' section, which contains three cards: 'Creative storytelling', 'Recipe creation', and 'Poetry generation'. At the bottom, there's a text input field for user queries and a note about token usage.

The screenshot shows the 'Add your data' configuration page. On the left, a sidebar lists steps: 1. Source data, 2. Index configuration, 3. Search settings, 4. Review and finish. Step 1 is highlighted with a blue circle. The main area shows 'Select your data' with a description and a 'Data source' dropdown set to 'Upload files'. A modal window titled 'Upload files' is open, showing options like 'Upload', 'Overwrite if already exists', and 'Upload list'. Below the modal, a message says 'No files uploaded' and provides instructions to select the Upload files or folders menu above to get started. At the bottom, there are 'Next', 'Create vector index', and 'Cancel' buttons.

4. Click on “Upload files” and “Upload folder”:

Add your data [PREVIEW](#)

- 1 Source data
- 2 Index configuration
- 3 Search settings
- 4 Review and finish

Select your data
 Select the data you want the generative AI to reference so it can ground its responses on your specific data.
 Your data will be ingested into an Index, which allows the Generative AI model to quickly and accurately find information for your specific use case.
Currently, only the file types .doc(x), .htm, .html, .md, .pdf, .ppt(x), .py, .txt, and .xls(x) are supported. Max file size limit is 16 MB.

Data source * ⓘ ▼

↗ Upload files File exists
 Upload list


No files uploaded
 Select the Upload files or folders menu above to get started.

ⓘ An Azure AI Search resource and an Azure Open AI connection will be required to index your data. [Create a new Azure AI](#) ▼

5. Select “product-info” folder. The Markdown files are uploaded to the blob storage.

Sélectionner le dossier à charger X

← → ⏪ ⏩ 📁 <> Lab LLMOps - GenAI_20241114 > product-info > ⏪ ⏩ Rechercher dans : product-i... 🔎

Organiser	Nom	Statut	Modifié le	Type
Accueil Franck – Microsoft Bureau Téléchargement Documents Images Musique Vidéos MS Gen AI Studio	📁 product-info	✓	13/11/2024 10:27	Dossier de fichier

Dossier :

Add your data PREVIEW

- 1 Source data
- 2 Index configuration
- 3 Search settings
- 4 Review and finish

Select your data
 Select the data you want the generative AI to reference so it can ground its responses on your specific data.
 Your data will be ingested into an Index, which allows the Generative AI model to quickly and accurately find information for your specific use case.
Currently, only the file types .doc(x), .htm, .html, .md, .pdf, .ppt(x), .py, .txt, and .xls(x) are supported. Max file size limit is 16 MB.

Data source * Upload files

Overwrite if already exists

Upload list

product-info/product_info_1.md	11.09 KB/11.09 KB	...
product-info/product_info_10.md	10.83 KB/10.83 KB	...
product-info/product_info_11.md	10.71 KB/10.71 KB	...
product-info/product_info_12.md	10.22 KB/10.22 KB	...
product-info/product_info_13.md	10.71 KB/10.71 KB	...
product-info/product_info_14.md	9.75 KB/9.75 KB	...

6. Select the Azure AI Search service from the drop-down menu and give your index a name. Leave “Auto select” for the virtual machine.

Add your data PREVIEW

- 1 Source data
- 2 Index configuration
- 3 Search settings
- 4 Review and finish

Index settings
 Configure your index

Index storage *
 Azure AI Search

Select Azure AI Search service * AzureAISearch

Vector index * product-info-lab2

Virtual machine * Auto select Select from recommended options Select from all options

Selecting a virtual machine will incur additional costs.

7. Click on “Add vector search to this search resource” and select an **Azure OpenAI connection** from the drop-down menu:

Add your data PREVIEW

- Source data
- Index configuration
- Search settings
- Review and finish

Configure search settings

Adding vector search supports: Hybrid (vector + keyword search), Hybrid + Semantic (most accurate search results for generative AI applications), Vector, Semantic and Keyword retrieval. Hybrid will be set as default and can be changed at inference time in the playground. Not adding vector search supports: Keyword and Semantic retrieval. Keyword will be set as default and can be changed at inference time in the playground. Adding vector search requires an Azure OpenAI embedding model. [Learn more](#)

Vector settings

Add vector search to this search resource

Azure OpenAI connection * (1)

ai-hublab2013149793759_aoai

(1) This resource requires an embedding model. If you don't have one already, **text-embedding-ada-002 (Version 2)** will be deployed for you. Using vector embeddings will incur usage to your account. [View Azure OpenAI Service pricing](#)

Back

Next

Create vector index

Cancel

8. Click on **Create vector index**:

Add your data PREVIEW

- Source data
- Index configuration
- Search settings
- Review and finish

Review and finish

Review the configurations you set for your index

Vector index

product-info-lab2

Index storage

Azure AI Search

Azure AI Search connection

AzureAISearch

Include vector settings

Yes

Schedule

OneTime

Compute

Serverless compute (Auto select)

Back

Create vector index

Cancel

9. After a few minutes, your index will be created. It includes vector embeddings.

✓ Add your data [PREVIEW](#)

Gain insights into your own data source. Your data is stored securely in your Azure subscription. [Learn more about how your data is protected.](#)

Index:
[product-info-lab2](#)

Search type:
Hybrid (vector + keyword) [Learn more about different search types](#)

Advanced settings >

 Remove data source

10. You can now chat with the model asking the question: "**How much are the TrailWalker hiking shoes?**", and this time it uses information from your data to construct the response. You can expand the **references** button to see the data that was used.

Chat playground

View code [Prompt flow](#) [Evaluate](#) Deploy [Launch](#) Import Export Prompt samples Send feedback

+ Add section

Clear chat Chat capabilities Show JSON

How much are the TrailWalker hiking shoes?

The TrailWalker Hiking Shoes are priced at \$110 ^1^ .
1 references
[product_info_11.md - Part 1](#)

Citations

Information about product item_number: 11

Information about product item_number: 11

Information about product item_number: 11

TrailWalker Hiking Shoes, price \$110

Brand

TrekReady

Category

Hiking Footwear

Features

- Durable and waterproof construction to withstand various terrains and weather conditions
- High-quality materials, including synthetic leather and mesh for breathability
- Reinforced toe cap and heel for added protection and durability
- Cushioned insole for enhanced comfort during long hikes
- Supportive midsole for stability and shock absorption
- Traction outsole with multidirectional lugs for excellent grip on different surfaces
- Breathable mesh lining to keep feet cool and dry
- Padded collar and tongue for extra comfort and to

Type user query here. (Shift + Enter for new line)

35/128000 tokens to be sent

Deploy your web app

Once you're satisfied with the experience in Azure AI Studio, you can deploy the model as a standalone web application. Publishing creates an Azure App Service in your subscription. It

might incur costs depending on the pricing plan you select. When you're done with your app, you can delete it from the Azure portal.

1. Select “Deploy...as a web app”:

The screenshot shows the Microsoft Bot Framework Chat playground. At the top, there's a toolbar with 'View code', 'Prompt flow', 'Evaluate', 'Deploy' (which is currently selected), 'Launch', 'Import', 'Export', 'Prompt samples', and 'Send feedback'. Below the toolbar, there are several sections: 'Add section', 'Add your data' (with a note about data protection), 'Deploy' dropdown (set to '...as a web app'), 'Show JSON' toggle, and 'Citations' panel. The 'Citations' panel lists 'Information about product item_number: 11' and 'Information about product item_number: 11' (both referring to TrailWalker Hiking Shoes). A central message card displays the response 'The TrailWalker Hiking Shoes are priced at \$110 ^1^ . 1 references [product_info_11.md - Part 1]'. On the right, there's a sidebar titled 'Citations' with the same information.

2. Fill in the different cells and click on Deploy:

The screenshot shows the 'Deploy to a web app' configuration dialog. It includes a note about Entra ID authentication and deployment time. Below, it asks to pick configurations for a new web app. The form fields are: Name (trek-app), Subscription (MCAPS-Hybrid-REQ-40894-2022-frgail), Resource group (labaistudio), Location (France Central), Pricing plan (Standard (\$1)), and a checkbox for 'Enable chat history in the web app'. At the bottom, there are 'View Pricing' and 'Deploy' (highlighted in blue) and 'Cancel' buttons.

3. Wait for the app to be deployed, which might take a few minutes.

Create a conversational RAG flow

Now you will create a conversational flow using the RAG pattern, start by creating a new flow in the **Prompt Flow** item in the **Tools** section within the **Build** tab.

Definition: Prompt flow is a development tool designed to streamline the entire development cycle of AI applications powered by Large Language Models (LLMs). Prompt flow provides a comprehensive solution that simplifies the process of prototyping, experimenting, iterating, and

deploying your AI applications. Prompt flow is available independently as an open-source project on [GitHub](#), with its own SDK and [VS Code extension](#).

Benefits of prompt flow:

With prompt flow in Azure AI Studio, you can:

- Orchestrate executable flows with LLMs, prompts, and Python tools through a visualized graph.
- Debug, share, and iterate your flows with ease through team collaboration.
- Create prompt variants and compare their performance.

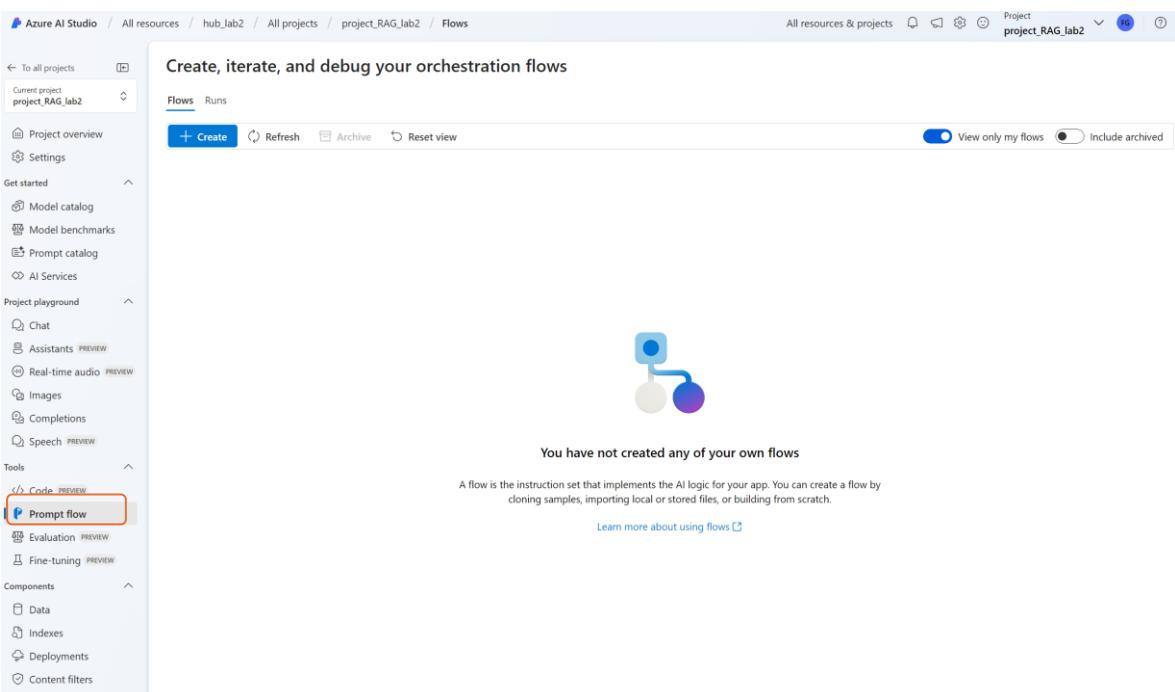
Enterprise readiness :

- *Collaboration:* Prompt flow supports team collaboration, allowing multiple users to work together on prompt engineering projects, share knowledge, and maintain version control.
- *All-in-one platform:* Prompt flow streamlines the entire prompt engineering process, from development and evaluation to deployment and monitoring. You can effortlessly deploy their flows as Azure AI endpoints and monitor their performance in real-time, ensuring optimal operation and continuous improvement.
- *Enterprise Readiness Solutions:* Prompt flow applies robust Azure AI enterprise readiness solutions, providing a secure, scalable, and reliable foundation for the development, experimentation, and deployment of flows.

With prompt flow in Azure AI Studio, you can unleash prompt engineering agility, collaborate effectively, and apply enterprise-grade solutions for successful LLM-based application development and deployment.

Action

1. Click on **Prompt Flow:**



2. Click on **Create** and **Clone** “Multi-Round Q&A on Your Data”:

The screenshot shows the 'Create a new flow' interface. At the top, there are three categories: 'Standard flow', 'Chat flow', and 'Evaluation flow'. Below these are sections for 'Explore gallery' and 'Import'. In the 'Explore gallery' section, there are eight cards arranged in two rows of four. The first card, 'Multi-Round Q&A on Your Data', is highlighted with a red oval and has its 'Clone' button highlighted with a blue rectangle. Other cards include 'Standard Q&A on Your Data', 'Standard Web Classification', 'Chat Chat with Wikipedia', 'Chat Use GPT Function Calling', 'Evaluation Classification Accuracy Eval...', 'Evaluation QnA Groundedness Evaluation', and 'Evaluation QnA Relevance Evaluation'. Each card has 'View detail' and 'Clone' buttons. On the right side of the 'Explore gallery' section, there is a link 'View more samples'. At the bottom right of the interface, there is a 'Cancel' button.

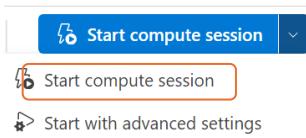
3. Start the automatic runtime by selecting **Start** in the **Runtime** drop-down. The runtime will be useful for you to work with the flow moving forward. The runtime provides the necessary computing resources for the flow to execute. This includes a Docker image with all the required dependencies and packages. The runtime ensures that the flow can run accurately and that any updates to the prompt or code content are properly integrated

The screenshot shows the Azure AI Studio interface with a flow named "Multi-Round Q&A on Your ...". The flow has two inputs: "chat_history" (list) and "chat_input" (string). The "chat_input" value is "How can I create one using azureml sdk V2?". The output is "chat_output" (\$chat_with_context.output). On the right, there is a "Graph" view showing the workflow: inputs -> modify_query_with_history -> lookup -> generate_prompt_context -> Prompt_variants -> chat_with_context. The "modify_query_with_history" node is expanded, showing its prompt template:

```

1 system:
2 * Given the following conversation history and the users next question, rephrase the question to be a stand alone que
3 If the conversation is irrelevant or empty, just restate the original question.
4 Do not add more details than necessary to the question.
5
6 chat history:
7 {% for item in chat_history %}
8 user:
9 {{ item.inputs.chat_input }}

```



Flow overview

The first node, **modify_query_with_history**, produces a search query using the user's question and their previous interactions.

Next, in the **lookup node**, the flow uses the vector index to conduct a search within a vector store, which is where the RAG pattern retrieval step takes place.

Following the search process, the **generate_prompt_context** node consolidates the results into a string.

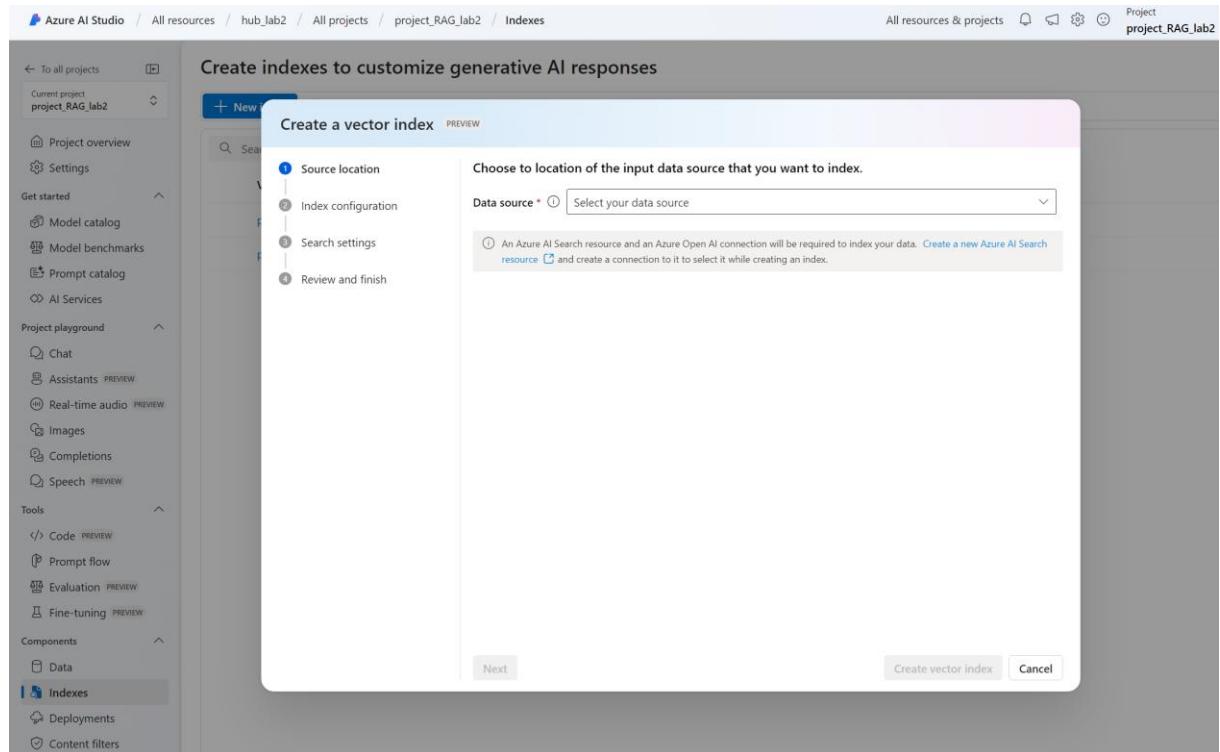
This string then serves as input for the **Prompt_variants** node, which formulates various prompts.

Finally, these prompts are used to generate the user's answer in the **chat_with_context** node.

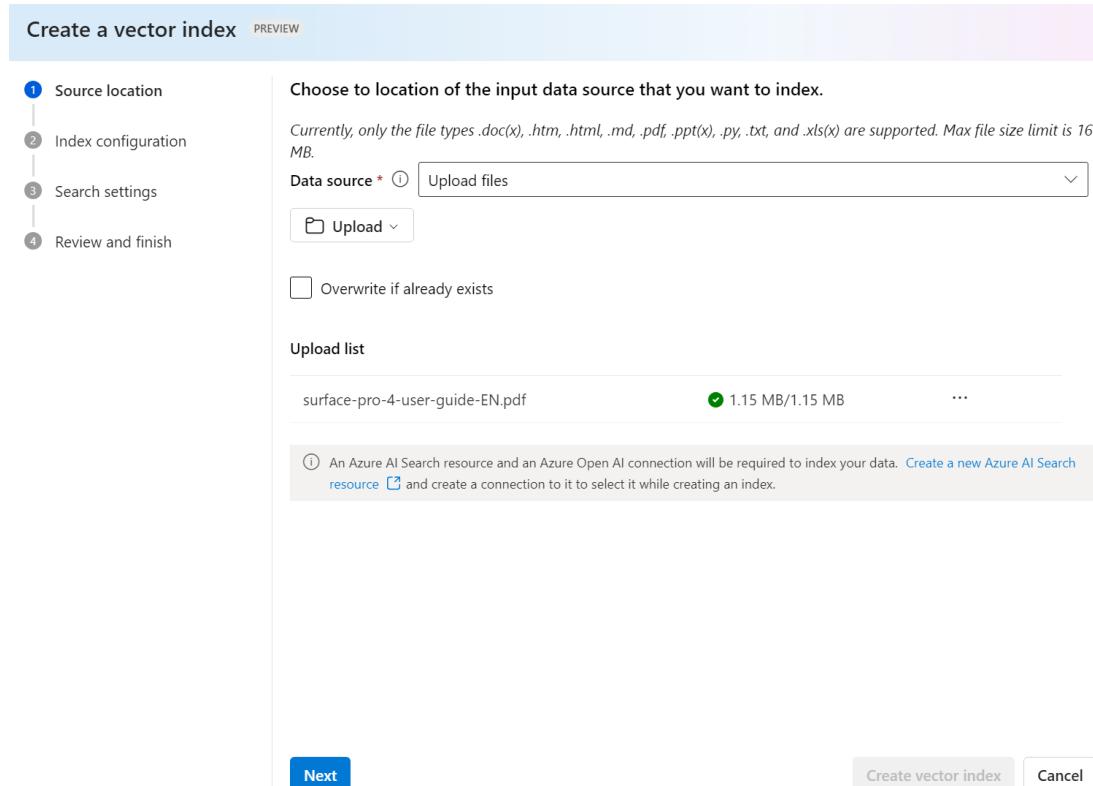
4. Create a search index

Before you can start running your flow, a crucial step is to establish the search index for the Retrieval stage. This search index will be provided by the Azure AI Search service.

In our case, we will create a **Vector index**. To do this, you just need to go back to the project in the **AI Studio**, select the **Indexes** option, and then click on the **New index** button.



At the **Data source** stage, select the **Upload files/folders** option and upload the **PDF files/surface-pro-4-user-guide-EN.pdf** to the data folder of this lab, as shown in the next screen.



In **Index storage**, select the Search Service you created earlier, give a name to your index and select “**Auto select**” for the virtual machine.

Create a vector index PREVIEW

1 Source location
2 Index configuration
3 Search settings
4 Review and finish

Index settings
Configure your index

Index storage *
Azure AI Search

Select Azure AI Search service * ⓘ
AzureAISearch

[Create a new Azure AI Search resource](#)

Vector index * ⓘ
surface-pro-index

Virtual machine * ⓘ
 Auto select Select from recommended options Select from all options

Selecting a virtual machine will incur additional costs.

Back **Next** **Create vector index** **Cancel**

Create a vector index PREVIEW

1 Source location
2 Index configuration
3 Search settings
4 Review and finish

Configure search settings
Combining hybrid retrieval with semantic ranking (Hybrid + Semantic) gives most accurate search results for generative AI applications. To generate vector index, embedding model is required.

Vector settings
 Add vector search to this search resource

Azure OpenAI connection * ⓘ
ai-hublab2013149793759_aoui

(ⓘ This resource requires an embedding model. If you don't have one already, [text-embedding-ada-002 \(Version 2\)](#) will be deployed for you. Using vector embeddings will incur usage to your account. [View Azure OpenAI Service pricing](#)

Back **Next** **Create vector index** **Cancel**

Click on **Create vector index**:

Create a vector index PREVIEW

- Source location
- Index configuration
- Search settings
- Review and finish

Review and finish

Review the configurations you set for your index

Vector index

surface-pro-index

Index storage

Azure AI Search

Azure AI Search connection

AzureAISearch

Include vector settings

Yes

Compute

Serverless compute (Auto select)

Back

Create vector index

Cancel

It may take about 10 minutes from the time it enters the execution queue until it starts.

surface-pro-index

Status

● Running

Version

-

Source type

Azure AI On Your Data

Total indexing time

-

Created on

Nov 14, 2024, 1:19:34 AM

↻ Refresh

Embed with model

No

Vector store

-

Compute

Serverless compute

Created by

Franck Gaillard

📄 Job details ⓘ

📄 Test data ⓘ

Status

Step 1 of 3
Cracking and chunking - Completed

Step 2 of 3
Creating Azure AI Search Index - Completed

Step 3 of 3
Registering Index - Not started

Source data

Name	Type	Size
surface-pro-4-user-guide-EN....	.pdf	1.15 MB

Wait until the index status is **Completed** as in the next image, before proceeding with the next steps.

surface-pro-index

Status

- Completed
- Version -
- Source type Azure AI On Your Data
- Total indexing time 8m
- Created on Nov 14, 2024, 1:19:34 AM

Embed with model No

Vector store -

Compute Serverless compute

Created by Franck Gaillard

Job details Test data

Source data

Name	Type	Size
surface-pro-4-user-guide-EN...	.pdf	1.15 MB

Status

- Step 1 of 3 Cracking and chunking - Completed ✓
- Step 2 of 3 Creating Azure AI Search Index - Completed ✓
- Step 3 of 3 Registering Index - Completed ✓

- Let's go back to the flow we cloned previously and let's configure the **lookup node**. After selecting the **lookup node**, click on **mlindex_content**.

lookup

Index Lookup Preview

Inputs

Name	Type	Value
mlindex_content	string	
queries	object	\$(modify_query_with_history.output)
query_type	string	
top_k	int	2

Activate config

A Generate window will appear. In this window, select the **Registered Index** option from the **index_type** field. Then, choose version 1 of the index you just created. After making these selections, click on **Save**.

Generate

Name	Type	Value
index_type	string	Registered Index
mlindex_asset_id	string	product-info-lab2:1

Now, let's go back to the **lookup node**. Select the **Hybrid (vector + keyword)** option from the **query_type** field.

- Now you will need to update the Connections of the nodes that link with LLM models. Starting with the Connection in the **modify_query_with_history** node with the **gpt-4o** deployment, as indicated below:

modify_query_with_history

Connection * ai-hublab2013149793759_aoui Api * chat

deployment_name * gpt-4o temperature 0 stop max_tokens 100

response_format Please choose an option

And the Connection for the **chat_with_context** node with the **gpt-4o** deployment, as indicated below:

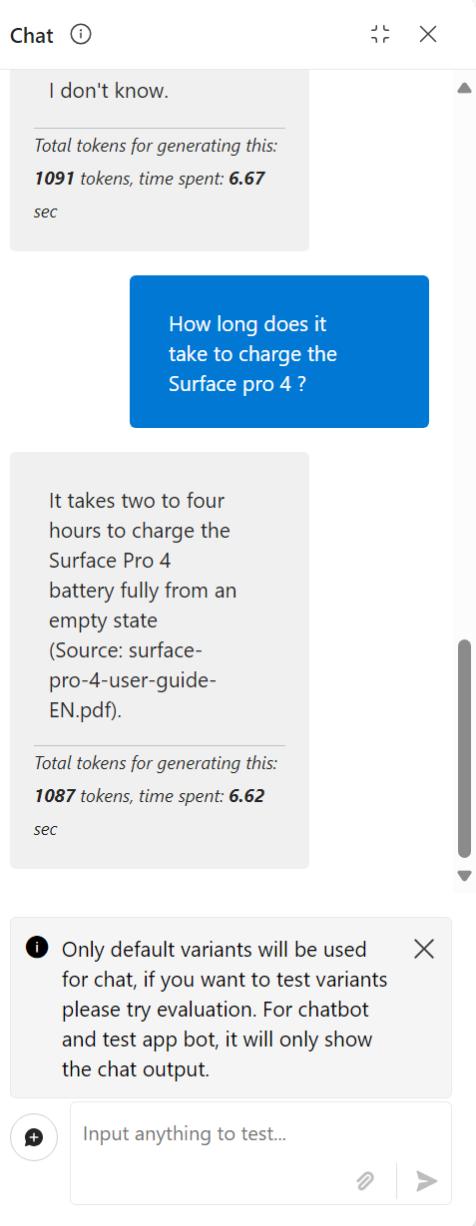
chat_with_context

Connection * ai-hublab2013149793759_aoui Api * chat

deployment_name * gpt-4o temperature 0 stop max_tokens 100

response_format Please choose an option

- Everything is now set up for you to initiate your chat flow. Simply click on the blue **Chat** button located at the top right corner of your page to begin interacting with the flow.



Evaluation

In this Lab, you will execute the following steps:

1. Evaluate your Chat flow.
2. Deploy the RAG flow to an online managed endpoint.

What is an evaluation?

An evaluation measures the quality and/or safety of a generative AI application. Designing and conducting an evaluation involves selecting metrics, creating test datasets, and undertaking iterative testing.

Action - Evaluate your Chat flow

Prepare you chat flow for evaluation

For the RAG flow that you created earlier to be evaluated, you must include additional information to the output node of this flow, specifically the context used to generate the answer.

This information will be used by the Evaluation Flow.

To do this, just follow these steps:

In the Flows section of **Prompt Flow**, open the **Multi-Round Q&A on Your Data** flow that you created in the previous lab. **This will be the flow we use for evaluation.**

Create, iterate, and debug your orchestration flows

The screenshot shows the Microsoft Flow 'Flows' list interface. At the top, there are tabs for 'Flows' (which is selected) and 'Runs'. Below the tabs are buttons for '+ Create', 'Refresh', 'Archive', and 'Reset view'. To the right are filters for 'View only my flows' and 'Include archived'. A search bar is present. The main area displays a table with columns: 'Flow name', 'Created by', 'Type', 'Created on', 'Updated on', and 'Description'. One row is visible, showing 'Multi-Round Q&A on Your Data-lab2' created by 'Franck Gaillard' as a 'Chat' type flow, created on Nov 13, 2024 at 10:22 PM, updated on Nov 14, 2024 at 1:36 AM, with the description 'Create a chatbot that uses LLM and data fr'.

Create a new output named **documents** in the **Outputs** node. This output will represent the documents that were retrieved in the **lookup node** and subsequently formatted in the **generate_prompt_context** node.

By creating a new output named **documents** and assigning the formatted documents to it, you're ensuring that the context used to generate answers is explicitly tracked. This context includes the documents retrieved and formatted during the flow.

Assign the output of the **generate_prompt_context** node to the **documents** output, as shown in the image below.

The screenshot shows the Microsoft Flow 'Outputs' configuration interface. It has a header with 'Name', 'Value', 'Chat output', and 'Action'. There are two entries: 'chat_output' with value '\${chat_with_context.output}' and 'Action' set to 'Selected'; and 'documents' with value '\${generate_prompt_context.output}' and 'Action' set to 'None'. Below the table is a button '+ Add output'.

Click **Save** before moving to the next section.

Create your evaluation flows

Still in the **Prompt flow** item in the **Tools** section, click on the blue **Create** button.

Create a new flow

Create by type

- Standard flow**: Harness the power of Large Language Models, customized Python code, and more to craft your tailored prompt flow. Test the flow using custom datasets and seamlessly deploy as an endpoint for easy integration.
[Create](#)
- Chat flow**: On top of the standard flow, this option provides the chat history support and a user-friendly chat interface in the authoring/debugging UI.
[Create](#)
- Evaluation flow**: Create an evaluation flow to measure how well the output matches the expected criteria and goals.
[Create](#)

Explore gallery

All Standard flow Chat flow Evaluation flow

Multi-Round Q&A on Your Data (Chat) Create a chatbot that uses LLM and data from your own indexed files to ground multi-round question and answering capabilities in enterprise chat scenarios.
[View detail](#) [Clone](#)

Q&A on Your Data (Standard) Use LLM and data from your own indexed files to ground multi-round question and answering capabilities.
[View detail](#) [Clone](#)

Web Classification (Standard) Use LLM to classify URLs into multiple categories.
[View detail](#) [Clone](#)

Chat with Wikipedia (Chat) Create a chatbot that leverages Wikipedia data to ground the responses.
[View detail](#) [Clone](#)

Use GPT Function Calling (Chat) Learn how to use GPT function calling to extend the capabilities of GPT models with external data sources.
[View detail](#) [Clone](#)

Classification Accuracy Evaluation (Evaluation) Measuring the performance of a classification system by comparing its outputs to groundtruth.
[View detail](#) [Clone](#)

QnA Groundedness Evaluation (Evaluation) Compute the groundedness of the answer for the given question based on the context.
[View detail](#) [Clone](#)

QnA Relevance Evaluation (Evaluation) Compute the relevance of the answer for the given question based on the context.
[View detail](#) [Clone](#)

Import

Cancel

Select the **Evaluation Flow** filter and click on **Clone** on the **QnA Groundedness Evaluation** card.

Explore gallery

All Standard flow Chat flow Evaluation flow

Classification Accuracy Evaluation (Evaluation) Measuring the performance of a classification system by comparing its outputs to groundtruth.
[View detail](#) [Clone](#)

QnA Groundedness Evaluation (Evaluation) **Clone** Compute the groundedness of the answer for the given question based on the context.
[View detail](#) [Clone](#)

QnA Relevance Evaluation (Evaluation) Compute the relevance of the answer for the given question based on the context.
[View detail](#) [Clone](#)

QnA Coherence Evaluation (Evaluation) Compute the coherence of the answer base on the question using llm.
[View detail](#) [Clone](#)

QnA Fluency Evaluation (Evaluation) Compute the Fluency of the answer base on the question using llm.
[View detail](#) [Clone](#)

QnA Ada Similarity Evaluation (Evaluation) Compute the cosine similarity between the answer and the ground truth embedded with ada embedding.
[View detail](#) [Clone](#)

QnA GPT Similarity Evaluation (Evaluation) Compute the similarity of the answer base on the question and ground truth using llm.
[View detail](#) [Clone](#)

QnA F1 Score Evaluation (Evaluation) Compute the F1 Score based on words in answer and ground truth.
[View detail](#) [Clone](#)

Clone flow

The flow code files are stored in a specific folder within your workspace file share storage. This folder name can be customized according to your preferences.

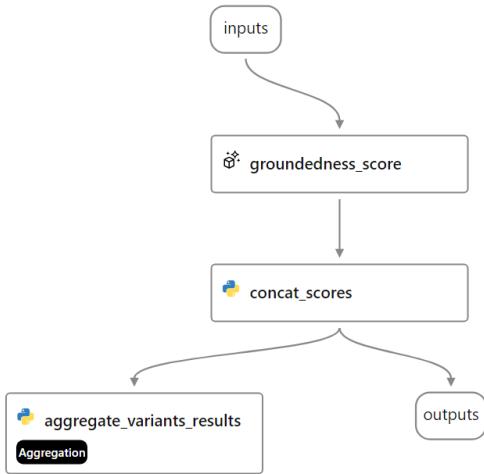
Location to store flow * ⓘ
Users/fgail/promptflow

Folder name * ⓘ
QnA Groundedness Evaluation-lab2

Clone Cancel

A flow will be created with the following structure:

Graph



Definition: The **Groundedness Evaluation** is a metric used to assess whether the responses generated by a generative AI model are based on the provided source materials. This is particularly important in Retrieval-Augmented Generation (RAG) scenarios, where the model retrieves and uses specific documents or data to generate answers.

Key Points:

- **Purpose:** To ensure that the AI's responses are accurate and based on the relevant data sources, rather than being fabricated or unrelated.
- **Process:** During evaluation, the model's responses are checked against the source documents to verify that the information provided is grounded in those documents.
- **Importance:** This helps in maintaining the reliability and trustworthiness of the AI model, ensuring that users receive accurate and contextually relevant information.

Update the **Connection** field to point to a **gpt-4o** deployment in **groundedness_score** node also update **max_tokens to 1000** as shown in the next figure.

groundedness_score

Connection: ai-hublab201349793759_aoai

deployment_name: gpt-4o

max_tokens: 1000

After updating the connection information, click on **Save** in the evaluation flow and navigate to the Flows section in **Prompt Flow** item.

Now, you will repeat the same steps described so far to create **two** additional evaluation flows, one **QnA Relevance Evaluation** and another **QnA GPT Similarity Evaluation**.

Explore gallery

All Standard flow Chat flow Evaluation flow

Evaluation Classification Accuracy Evaluation Measuring the performance of a classification system by comparing its outputs to groundtruth. View detail Clone	Evaluation QnA Groundedness Evaluation Compute the groundedness of the answer for the given question based on the context. View detail Clone	Evaluation QnA Relevance Evaluation Compute the relevance of the answer for the given question based on the context. View detail Clone	Evaluation QnA Coherence Evaluation Compute the coherence of the answer base on the question using llm. View detail Clone
Evaluation QnA Fluency Evaluation Compute the Fluency of the answer base on the question using llm. View detail Clone	Evaluation QnA Ada Similarity Evaluation Compute the cosine similarity between the answer and the ground truth embedded with ada embedding. View detail Clone	Evaluation QnA GPT Similarity Evaluation Compute the similarity of the answer base on the question and ground truth using llm. View detail Clone	Evaluation QnA F1 Score Evaluation Compute the F1 Score based on words in answer and ground truth. View detail Clone

Definition: The **relevance score** is a metric used to measure how well the AI model's responses match the user's query or the intended context. Here's a simplified explanation:

Key Points:

- **Purpose:** To evaluate the quality and appropriateness of the AI-generated responses.
- **Calculation:** The score is computed based on how closely the response aligns with the user's query and the provided context. *The higher the score, the more relevant the response is considered to be.*
- **Usage:** This metric helps in assessing and improving the performance of generative AI models by ensuring that the responses are not only accurate but also contextually appropriate

In essence, the **relevance score** helps you understand how effectively your AI model is meeting user needs by providing pertinent and useful answers.

Update the **Connection** field to point to a **gpt-4o** deployment in **relevance_score** node also update **max_tokens** to **1000** as shown in the next figure.



Definition: The **similarity score** is a metric used to measure how closely the AI model's generated responses match the expected or reference responses. Here's a simplified explanation:

Key Points:

- **Purpose:** To evaluate the degree of similarity between the AI-generated text and the reference text.
- **Calculation:** This score is typically calculated using various methods, such as cosine similarity, which compares the semantic meaning of the texts by converting them into vector representations

- **Usage:** It helps in assessing how well the AI model is performing in terms of generating responses that are semantically similar to the expected answers. This is particularly useful for tasks like text generation, translation, and summarization

In essence, the **similarity score** helps ensure that the AI model's outputs are not only relevant but also closely aligned with the intended responses, improving the overall quality and reliability of the model.

Update the **Connection** field to point to a **gpt-4o** deployment in **similarity_score** node also update **max_tokens to 1000** as shown in the next figure.

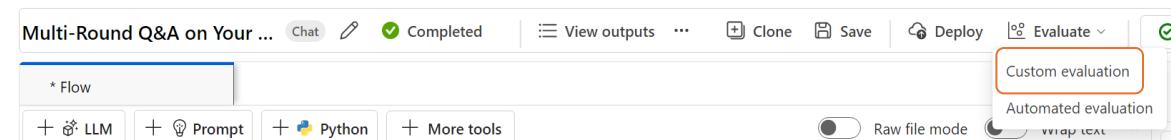


Run the evaluation

In the Flows section of **Prompt Flow**, open the **Multi-Round Q&A on Your Data** flow that you created in the previous lab. This will be the flow we use for evaluation.

Start the automatic runtime by selecting **Start** in the **Runtime** drop-down.

Select the **Custom evaluation** option in the Evaluate menu.



Prompt variants refer to different versions of a prompt or tool node that have distinct settings.

In the **Prompt_variants** option, select the option to run only **two variants** to avoid reaching your GPT-4o model quota limit, as shown in the example image below.

Batch run & Evaluate

Basic settings

- Run display name *
- Run description
- Tags

Key	Value
+ Add tag	
- Variants ***
Select a node with variants that you want to run. Note: other nodes will run with default variant.
 Select a node to run variants Use default variant for all nodes

Node name	Variant id
Prompt_variants	variant_0(default),variant_1
(1) 2 run(s) will be generated based on selected variant(s)	<input type="checkbox"/> (Select all) <input checked="" type="checkbox"/> variant_0(default) <input checked="" type="checkbox"/> variant_1 <input type="checkbox"/> variant_2

Review + submit

Select **Add new data**.

Batch run & Evaluate

Basic settings (checked)

- Batch run settings
- Evaluation settings
- Select evaluation optional
- Configure evaluation optional
- Review

Batch run settings

Data *

Selected file must be **.jsonl, .csv, .tsv, or a folder containing these types.**

- Select a **.jsonl, .csv, or .tsv file, or a folder containing these file types.**

+ Add new data

Add new data

Name *

Data with same name will be saved as a new version

Upload from local file Upload from local folder

Choose a file *

Please make sure the data includes headers

Input mapping *

Input	Type
chat_history	list
chat_input	string

Cannot preview

Review + submit

Upload the file **data.csv**.

Add new data

X

Name *

testdata

Data with same name will be saved as a new version

Upload from local file Upload from local folder

Choose a file *

data.csv

 Browse

Please make sure the data includes headers

After clicking on **Add** proceed to map the input fields as shown below:

Batch run & Evaluate

X

- Basic settings
- Batch run settings
- Evaluation settings
 - Select evaluation optional
 - Configure evaluation optional
- Review

Batch run settings

Data * 

testdata (version 1)

 Selected file must be jsonl, csv, .tsv, or a folder containing these types.

• Select a jsonl, csv, or .tsv file, or a folder containing these file types.

[+ Add new data](#)

Input mapping *

Input	Type	Dataset column
chat_history	list	<code>\$(data.chat_history)</code> 
chat_input	string	<code>\$(data.question)</code> 

Preview of top 5 rows

chat_history	question	answer	documents
[1]	What does Windows 10 Provides?	Windows 10 offers a variety of new featu...	Windows 10 provides new features and ...
[1]	How much RAM does Surface Pro 4 can ...	Surface Pro 4 is available with up to 16 ...	Memory and storage Surface Pro 4 is av...
[1]	How do I check the battery level on my ...	You can check the battery level from the ...	Check the battery level You can check th...
[1]	What processor does the Surface Pro 4 h...	The Surface Pro 4 is equipped with a 6th...	Processor The 6th-generation Intel Core ...
[1]	Can I use a pen with the Surface Pro 4?	Yes; the Surface Pro 4 comes with the Su...	Surface Pen Enjoy a natural writing exper...

[Previous](#)

[Next](#)

[Review + submit](#)

[Cancel](#)

Select the three evaluation flows you just created.

Batch run & Evaluate

Select evaluation

You can choose to test your prompt flow and evaluate the output performance using automated or customized evaluation method. You can submit batch run without evaluation if you want to evaluate the outputs later.

Customized evaluation

3 evaluation(s) selected [Remove all](#)

- QnA Relevance Evaluation-lab2 [View details](#)
- QnA GPT Similarity Evaluation-lab2 [View details](#)
- QnA Groundedness Evaluation-lab2 [View details](#)

Automated evaluation

0 evaluation(s) selected [Remove all](#)

- Classification Accuracy Evaluation [View details](#)
- QnA Groundedness Evaluation [View details](#)
- QnA Relevance Evaluation [View details](#)
- QnA Coherence Evaluation [View details](#)
- QnA Fluency Evaluation [View details](#)
- QnA Ada Similarity Evaluation [View details](#)

[Previous](#) [Next](#) [Review + submit](#) [Cancel](#)

Click on **Next** to set up the **question**, **context**, **ground_truth** and **answer** fields for each evaluation flow. You can see how to do this in the three images below.

QnA GPT Similarity Evaluation-lab2 [Apply to all evaluations](#)

Evaluation input mapping *

Choose data asset for evaluation * ⓘ

testdata (version 1)

[+ Add new data](#)

Name	Description	Type	Data source
question		string	<code>\$(data.question)</code> Edit
ground_truth		string	<code>\$(data.answer)</code> Edit
answer		string	<code>\$(run.outputs.chat_output)</code> Edit

QnA Groundedness Evaluation-lab2 [Apply to all evaluations](#)

Evaluation input mapping *

Choose data asset for evaluation * ⓘ

testdata (version 1)

[+ Add new data](#)

Name	Description	Type	Data source
question		string	<code>\$(data.question)</code> Edit
context		string	<code>\$(data.answer)</code> Edit
answer		string	<code>\$(run.outputs.chat_output)</code> Edit

QnA Relevance Evaluation-lab2 ✓

Apply to all evaluations

Evaluation input mapping *

Choose data asset for evaluation * ⓘ

testdata (version 1)

+ Add new data

Name	Description	Type	Data source
question		string	\$(data.question)
context		string	\$(data.answer)
answer		string	\$(run.outputs.chat_output)

Click on **Submit** to start the evaluation.

The evaluation process has started. To view all evaluations (one per variant), please navigate to the **Evaluation** section under the **Build** tab.

The screenshot shows the Azure AI Studio interface with the 'Evaluation' tab selected. On the left, there's a sidebar with various project and tool options. The main area is titled 'Assess and compare AI application performance' and contains a step-by-step guide for evaluation. A table below lists completed evaluations, each with a status of 'Completed' and a score of 4.9. An orange oval highlights this table.

Evaluations	Status	Created on	Groundedness	Relevance	Retrieval score	Coherence	Similarity
Multi-Round Q&A on Your Data-11-13-2024-22-20-	Completed	Nov 14, 2024 2:50 AM	4.2	--	--	--	--
Multi-Round Q&A on Your Data-11-13-2024-22-20-	Completed	Nov 14, 2024 2:50 AM	3.6	--	--	--	--
Multi-Round Q&A on Your Data-11-13-2024-22-20-	Completed	Nov 14, 2024 2:50 AM	--	--	--	--	4.25
Multi-Round Q&A on Your Data-11-13-2024-22-20-	Completed	Nov 14, 2024 2:50 AM	--	--	--	--	4.65
Multi-Round Q&A on Your Data-11-13-2024-22-20-	Completed	Nov 14, 2024 2:50 AM	--	4.95	--	--	--
Multi-Round Q&A on Your Data-11-13-2024-22-20-	Completed	Nov 14, 2024 2:50 AM	--	4.9	--	--	--

Groundedness

- Use it when:** You're worried your application generates information that isn't included as part of your generative AI's trained knowledge (also known as unverifiable information).|
- How to read it:** If the model's answers are highly grounded, it indicates that the facts covered in the AI system's responses are verifiable by the input source or internal database. Conversely, low groundedness scores suggest that the facts mentioned in the AI system's responses may not be adequately supported or verifiable by the input source or internal database. In such cases, the model's generated answers could be based solely on its pretrained knowledge, which may not align with the specific context or domain of the given input

- **Scale:**

- 1 = "ungrounded": suggests that responses aren't verifiable by the input source or internal database.
- 5 = "perfect groundedness" suggests that the facts covered in the AI system's responses are verifiable by the input source or internal database.

Relevance

- **Use it when:** You would like to achieve high relevance for your application's answers to enhance the user experience and utility of your generative AI systems.
- **How to read it:** Answers are scored in their ability to capture the key points of the question from the context in the ground truth source. If the model's answers are highly relevant, it indicates that the AI system comprehends the input and can produce coherent and contextually appropriate outputs. Conversely, low relevance scores suggest that the generated responses might be off-topic, lack context, or fail to address the user's intended queries adequately.
- **Scale:**
 - 1 = "irrelevant" suggests that the generated responses might be off-topic, lack context, or fail to address the user's intended queries adequately.
 - 5 = "perfect relevance" suggests contextually appropriate outputs.

Similarity

- **Use it when:** You would like to objectively evaluate the performance of an AI model (for text generation tasks where you have access to ground truth desired responses). Ada similarity allows you to compare the generated text against the desired content.
- **How to read it:** Answers are scored for equivalencies to the ground-truth answer by capturing the same information and meaning as the ground-truth answer for the given question. A high Ada similarity score suggests that the model's prediction is contextually similar to the ground truth, indicating accurate and relevant results. Conversely, a low Ada similarity score implies a mismatch or divergence between the prediction and the actual ground truth, potentially signaling inaccuracies or deficiencies in the model's performance.
- **Scale:**
 - 1 = "nonequivalence" suggests a mismatch or divergence between the prediction and the actual ground truth, potentially signaling inaccuracies or deficiencies in the model's performance.
 - 5 = "perfect equivalence" suggests that the model's prediction is contextually similar to the ground truth, indicating accurate and relevant results.

Note: you can use the SDK to evaluate your models => [Evaluate with the Azure AI Evaluation SDK - Azure AI Studio | Microsoft Learn](#)

Deploy the RAG flow to an online managed endpoint

Open the **Multi-Round Q&A on Your Data** flow that you created in the previous lab.

Now that you have built a flow and tested it properly, it's time to create your online endpoint for real-time inference.

Follow the steps below to deploy a prompt flow as an online endpoint in Azure AI Studio.

1. Have a prompt flow ready for deployment.
2. Select **Deploy** on the flow editor.

Deploy Multi-Round Q&A on Your Data-lab2

1 Basic settings

2 Advanced settings

3 Review

Basic settings

Deploy your flow to a managed online endpoint for real-time inference. [Learn more](#)

Endpoint

New Existing

Endpoint name * [\(i\)](#)
project-rag-lab2-endpoint

Deployment name * [\(i\)](#)
project-rag-lab2-endpoint-1

Virtual machine * [\(i\)](#)
Standard_DS3_v2 4 Cores, 14 GB (RAM), 28 GB (Disk), \$0.29/hr

Instance count * [\(i\)](#)
3

Inferencing data collection [\(i\)](#)
 Enabled

Deploy Multi-Round Q&A on Your Data-lab2

1 Basic settings

2 Advanced settings

3 Review

Review the deployment settings

Basic settings

Endpoint name
project-rag-lab2-endpoint

Deployment Name
project-rag-lab2-endpoint-1

Virtual machine
Standard_DS3_v2

Instance count
3

Inferencing data collection
Enabled

Application Insights diagnostics
Disabled

Endpoint

Authentication type
Key

Public network access
Enabled

Description
--

Identity type
system

Enforce access to connection secrets (preview)
Enabled

Endpoint tags

Deployment

Tags
[\(i\)](#) No tags

Environment
Use environment of current flow definition

Outputs

Output name	Type
chat_output	string
documents	string

Connection

Node name	Provider	Connection	Deployment name / Model
chat_with_context	AzureOpenAI	ai-hublab201314979_3759_aoui	gpt-4o
modify_query_with_history	AzureOpenAI	ai-hublab201314979_3759_aoui	gpt-4o

Create **Back** **Cancel**

Manage deployments of your models, apps, and services

Deploy a model with your private API key and an endpoint URI (Uniform Resource Identifier).

Model deployments App deployments Service deployments

Name	Model name	Model version	State	Model retirement date	Content filter	Deployment ty
ai-hublab2013149793759_aoai	Azure OpenAI					
gpt-4o	gpt-4o	2024-05-13	Succeeded		DefaultV2 ⓘ	Global Standard
text-embedding-3-large	text-embedding-3-large	1	Succeeded		DefaultV2 ⓘ	Standard
text-embedding-ada-002	text-embedding-ada-002	2	Succeeded		Default ⓘ	Standard
project-rag-lab2-endpoint	Endpoint					
project-rag-lab2-endpoint-1	project-rag-lab2-endpoint		Succeeded			

You can test it:

project-rag-lab2-endpoint-1

Details Test Consume Monitoring PREVIEW Logs

Chat mode

how long does it take to charge the surface pro4?

It takes two to four hours to charge the Surface Pro 4 battery fully from an empty state (Source: surface-pro-4-user-guide-EN.pdf).

Discover Content Safety

Responsible AI Principles



The idea behind Microsoft's Responsible AI Principles is to ensure that AI technologies are developed and used in ways that are ethical, trustworthy, and beneficial to society. These principles guide the design, deployment, and governance of AI systems to address potential risks and maximize positive impacts.

Microsoft's responsible AI framework is built on 6 key principles:

1. **Fairness:** Ensuring AI systems treat all users equitably.
2. **Reliability and Safety:** Guaranteeing that AI systems function as intended and are safe to use.
3. **Privacy and Security:** Protecting user data and ensuring confidentiality.
4. **Inclusiveness:** Making AI accessible and beneficial to everyone.
5. **Transparency:** Being open about how AI systems work and make decisions.
6. **Accountability:** Taking responsibility for AI systems.

Content Filtering System



The content filtering system is a practical implementation of these principles:

1. Fairness:

- Filters are designed to detect and mitigate biases in content, ensuring equitable treatment of all users.

2. Reliability and Safety:

- The system prevents harmful outputs by filtering content related to violence, hate speech, explicit material, and self-harm, ensuring the AI's reliability and user safety.

3. Privacy and Security:

- Filters help protect sensitive information and prevent the dissemination of private data.

4. Inclusiveness:

- By filtering out harmful content, the system promotes a safe and inclusive environment for all users.

5. Transparency:

- Users are informed about the filtering mechanisms and can adjust settings to suit their needs, promoting transparency in how the AI operates.

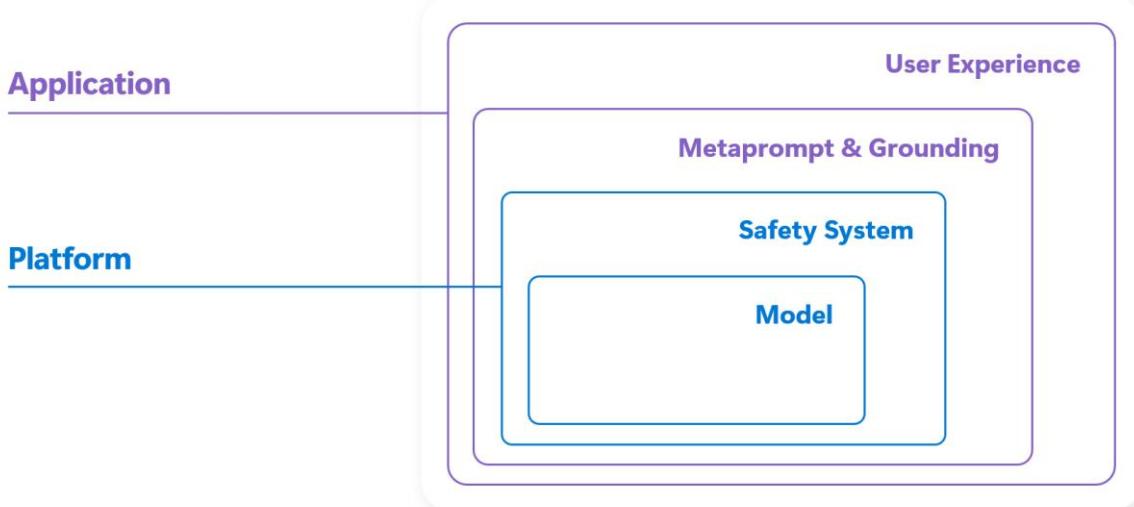
6. Accountability:

- Microsoft continuously monitors and updates the filtering system to address new challenges and ensure it aligns with ethical standards, demonstrating accountability.

Mitigation Layers

The concept of mitigation layers is applied to ensure the safe, ethical, and effective use of AI models. Here's how each layer functions within this framework:

Mitigation layers



The Model layer

The **Model** layer is where the AI processes inputs and generates outputs. In Azure OpenAI, this involves advanced neural networks like GPT-4. To ensure the outputs are safe and appropriate, Azure OpenAI employs a robust content filtering system.

The content filtering system in Azure OpenAI works alongside the core models to detect and prevent harmful content.

[Here's how it functions:](#)

1. Multi-Class Classification Models:

- These models analyze both the input prompts and the output completions.
- They are trained to detect harmful content across four main categories: **violence, hate, sexual, and self-harm**.

2. Severity Levels:

- Content is classified into four severity levels: **safe, low, medium, and high**.
- By default, content detected at medium or high severity levels is filtered out, while content at low or safe levels is not.

3. Categories of Harmful Content:

- **Hate:** Content that promotes hate speech or discrimination.
- **Sexual:** Explicit or inappropriate sexual content.
- **Violence:** Content related to physical harm or threats.
- **Self-Harm:** Content that promotes self-injury or suicide.

4. Optional Filters:

- **Jailbreak Risk Detection:** Identifies attempts to bypass safety mechanisms.

- [**Protected Material Detection:** Flags known text or code from public repositories to prevent unauthorized use.](#)

How It Works

- **Input and Output Analysis:** Both the user's input and the AI's output are analyzed by the classification models.
- **Filtering Actions:** If harmful content is detected at medium or high severity levels, the system filters it out. For lower severity levels, content may be annotated but not necessarily blocked.

Configurability

- Users can configure the content filters to adjust the severity thresholds for different categories. [This allows for customization based on specific use cases and requirements.](#)

The Safety System

This content filtering system is powered by [Azure AI Content Safety](#), and it works by running both the prompt input and completion output through an ensemble of classification models aimed at detecting and preventing the output of harmful content. Variations in API configurations and application design might affect completions and thus filtering behavior.

With Azure OpenAI model deployments, **you can use the default content filter or create your own content filter** (described later on). The default content filter is also available for other text models curated by Azure AI in the [model catalog](#), but **custom content filters aren't yet available for those models**.

Models available through **Models as a Service** have content filtering enabled by default and can't be configured.

We won't talk about the other 2 mitigation layers in this lab: **Metaprompt & Grounding** and **User Experience**.

Action

- First, let's test the behavior of the Mistral Large 24-07 model, select the **Project Playground** option and the **Chat** option and copy the following prompt in the "**Give the model instructions and context**" cell:

You're an AI assistant that helps telco company to extract valuable information from their conversations by creating JSON files for each conversation transcription you receive.

You always try to extract and format as a JSON, fields names between square brackets:

1. Customer Name [name]
2. Customer Contact Phone [phone]
3. Main Topic of the Conversation [topic]
4. Customer Sentiment (Neutral, Positive, Negative)[sentiment]
5. How the Agent Handled the Conversation [agent_behavior]
6. What was the FINAL Outcome of the Conversation [outcome]
7. A really brief Summary of the Conversation [summary]

← Chat playground ▾

</> View code

Setup Hide

Deployment * Create new deployment

Mistral-large-2407-labai

Give the model instructions and context ⓘ

[agent_behavior]
6. What was the FINAL Outcome of the Conversation
[outcome]
7. A really brief Summary of the Conversation [summary]
Only extract information that you're sure. If you're unsure, write "Unknown/Not Found" in the JSON file.

Apply changes Generate prompt ↻

+ Add section ▾

Safety system messages

Examples

Variable

- Click on **Safety system messages** and check “**Avoid harmful content**”:

Select safety system message(s) to insert

Insert one or more prepared system messages into your prompt; you can alter or add to them if you'd like. Token usage will be incurred when you begin chatting with the model in the playground.

- Select all (276 tokens)
- Avoid harmful content (61 tokens)
- Avoid ungrounded content (93 tokens)
- Avoid copyright infringements (81 tokens)
- Avoid jailbreaks and manipulation (41 tokens)

Insert

Cancel

- Copy in the prompt this conversation and click on **Apply changes**:

Agent: Hi Mr. Perez, welcome to Telco's customer service. My name is Juan, how can I assist you?

Client: Hello, Juan. I am very dissatisfied with your services.

Agent: ok sir, I am sorry to hear that, how can I help you?

Client: I hate this company I will kill everyone with a bomb.

- Check the response from Mistral Large-2407, the Violence filter was triggered with the text.

The screenshot shows the Azure AI Foundry interface for a project named 'axway-genai'. The left sidebar includes sections like Overview, Model catalog, Playgrounds (selected), AI Services, Build and customize, Code, Fine-tuning, Prompt flow, Assess and improve, Tracing, Evaluation, Safety + security, My assets, Models + endpoints, Data + indexes, and Web apps. The main area is titled 'Chat playground' and contains a 'Setup' section with a deployment dropdown set to 'Mistral-large-2407-labai'. Below it is a 'Give the model instructions and context' section with numbered prompts. A 'Safety system message' section follows, containing rules for avoiding harmful content. The right side features a 'Chat session' pane with a conversation between an Agent and a Client. The Client's message 'I will kill everyone with a bomb.' was filtered due to Microsoft's content management policy. A pink box at the bottom of the session pane states: 'The response was filtered due to the prompt triggering Microsoft's content management policy. Please modify your prompt and retry.'

- “Bomb” has been detected as harmful content, and the reply has been filtered.