Personalised AI

Azure OpenAI on data allows developers to use supported AI chat models that can reference specific sources of data to ground the response. Adding this information allows the model to reference both the specific data provided and its pretrained knowledge to provide more effective responses. Azure OpenAI on personalized data utilizes the search ability of Azure Cognitive Search to add the relevant data chunks to the prompt.

Once the data is in a Cognitive Search Index, Azure OpenAI on personalized data goes through the following steps:

1. Receive user prompt.
2. Determine relevant content and intent of the prompt.
3. Query the search index with that content and intent.
4. Insert search result chunk into the Azure OpenAI prompt, along with system message and user prompt.
5. Send the entire prompt to Azure OpenAI.
6. Return response and data reference (if any) to the user.

By default, Azure OpenAI on personalised data, encourages, but doesn't require the model to respond only using personalised data. This setting can be unselected when connecting data, which may result in the model choosing to use its pretrained knowledge over personalised data.

Fine tuning VS personalised data

Fine-tuning is a technique used to create a custom model by training an existing foundational model such as gpt-35-turbo with a dataset of additional training data. Fine-tuning can result in higher quality requests that prompt engineering alone, customize the model on examples larger than can fit in a prompt, and allow user to provide fewer examples to get the same high-quality response. However, the process for fine-tuning is both costly and time intensive and should only be used for *use cases* where it's necessary.

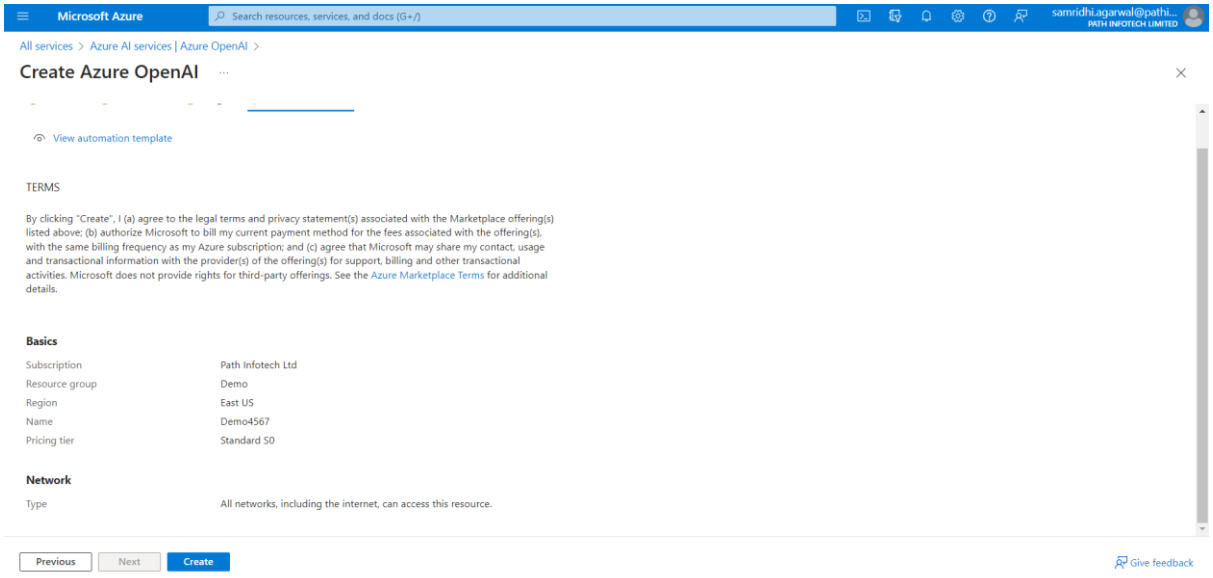
Azure OpenAI on personalized data still uses the stateless API to connect to the model, which removes the requirement of training a custom model with personalized data and simplifies the interaction with the AI model. Cognitive Search first finds the useful information to answer the prompt, then Azure OpenAI forms the response based on the information.

Adding data is done through the Azure AI Studio, in the Chat Playground. The data source added is then used to augment the prompt sent to the model. When adding data one can choose to upload data files, use data in blob storage account, or connect to an existing Cognitive Search index. If using large text files or forms, use the available data preparation script to improve the AI model's accuracy.

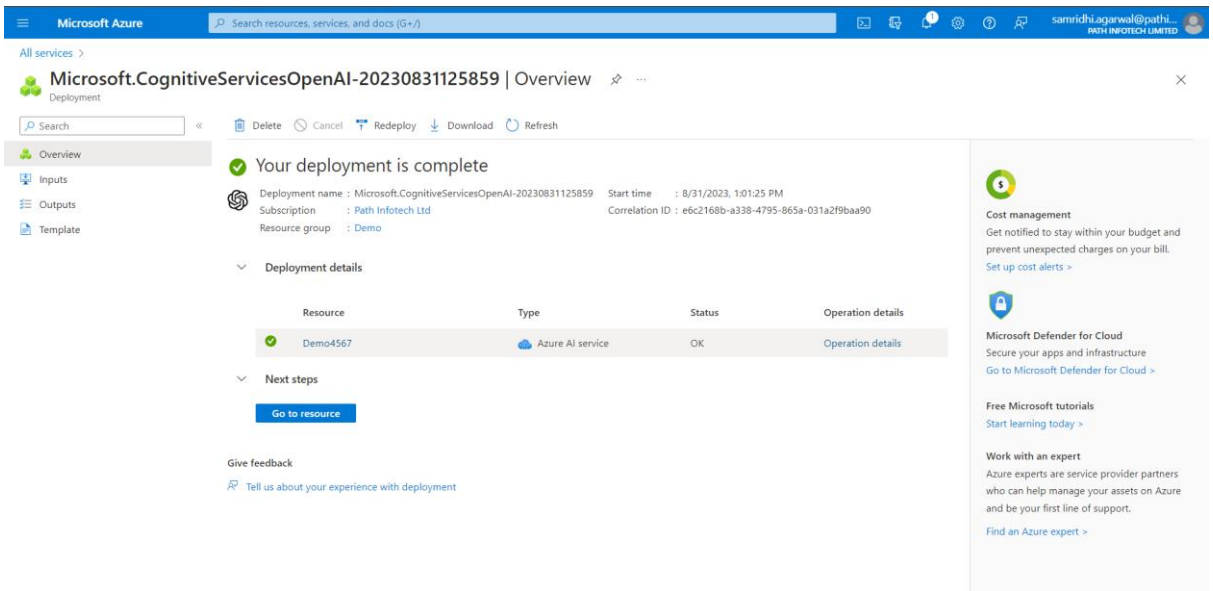
Note: If uploading or using files already in the storage account, Azure OpenAI supports .md, .txt, .html, .pdf, .docx and .pptx. If any of these files contain graphics or images, the response quality depends on how well text can be extracted from the visual content.

STEPS

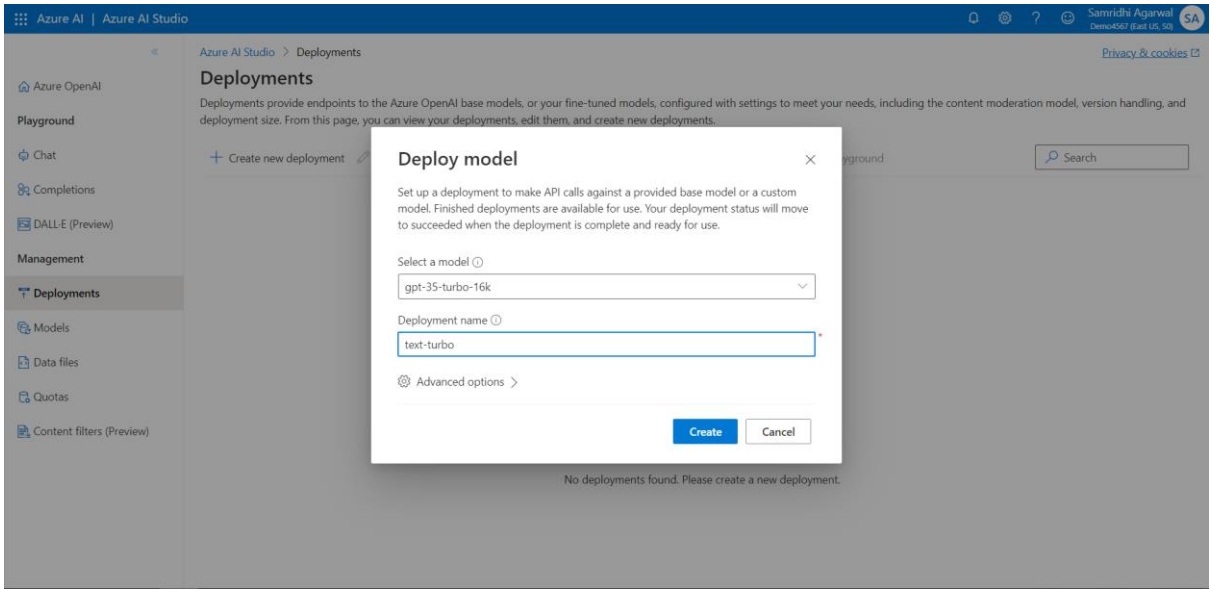
1. Provision an Azure OpenAI resource with the following credentials:
    i. Subscription: Path InfotechLtd
    ii. Resource group: Demo
    iii. Region: EastUS
    iv. Name: Demo4567
    v. Pricing: Standard S0



Wait for the deployment to complete then go to the deployed Azure OpenAI resource in Azure portal.



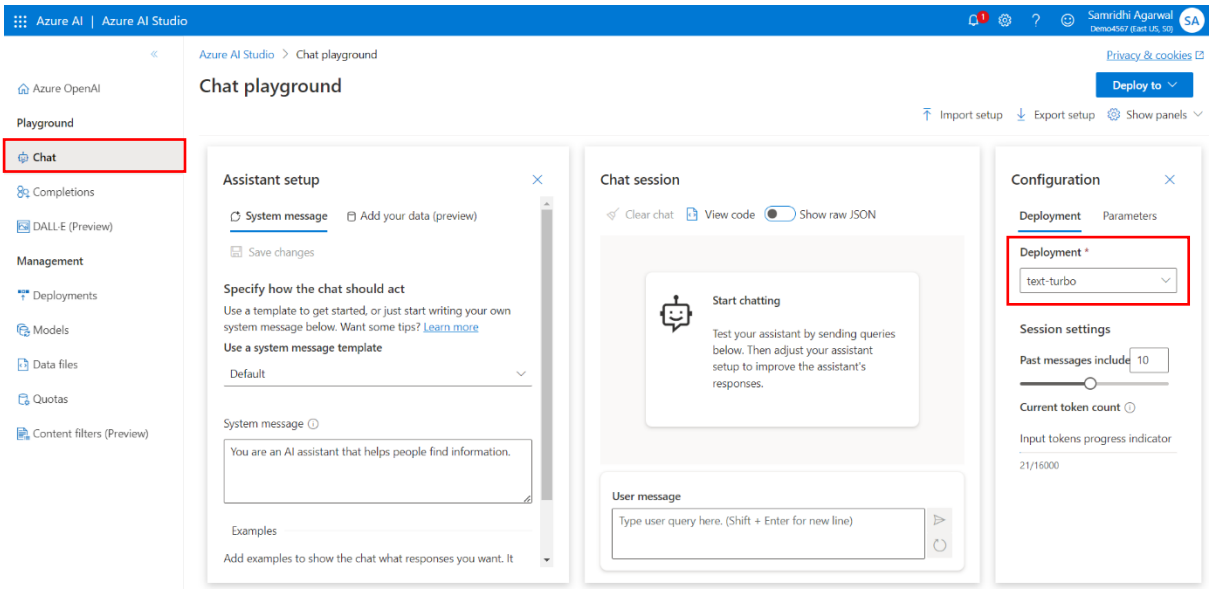
2. Deploying a model
   To use Azure OpenAI API for code generation, a model must be deployed first to use through the Azure OpenAI Studio. Once deployed, this model is used with the playground and reference that model in the app.
   Deploy a model with the following credentials:
   i. Model: gpt-35-turbo-16k
   ii. Name: turbo-text



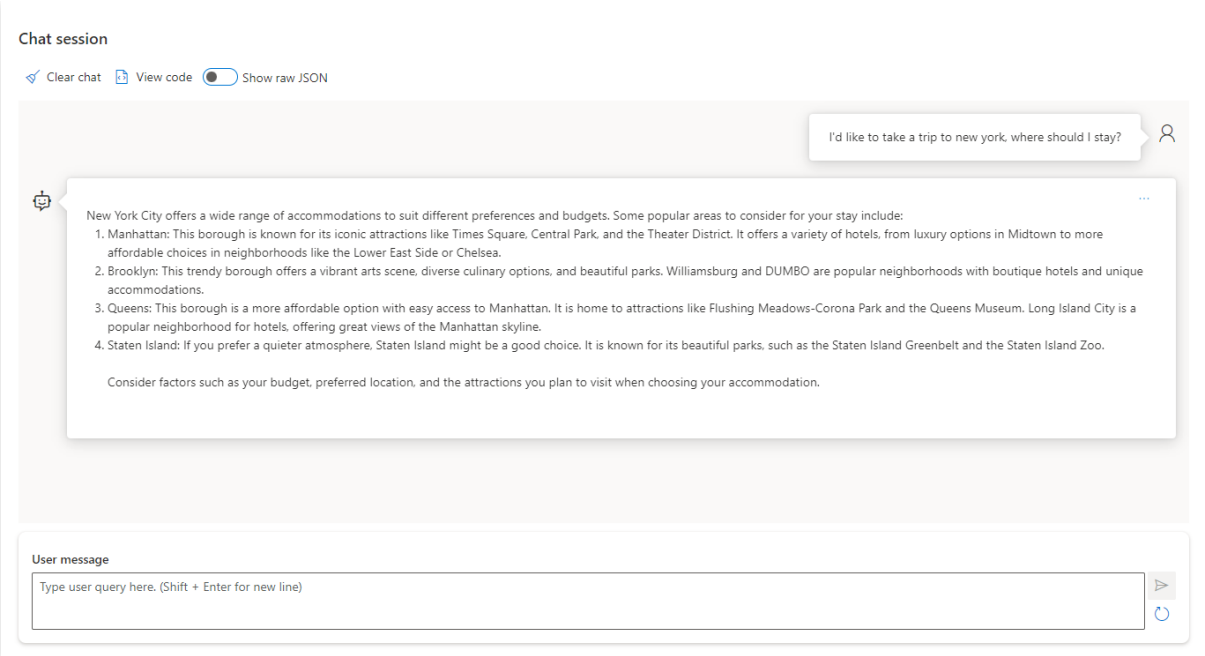
Observing normal chat behavior without adding your own data

Before connecting personalized data to Azure OpenAI, let's observe the base model response to queries without any grounding data.

1. Navigate to the Chat playground, and select the deployed model in the configuration pane.

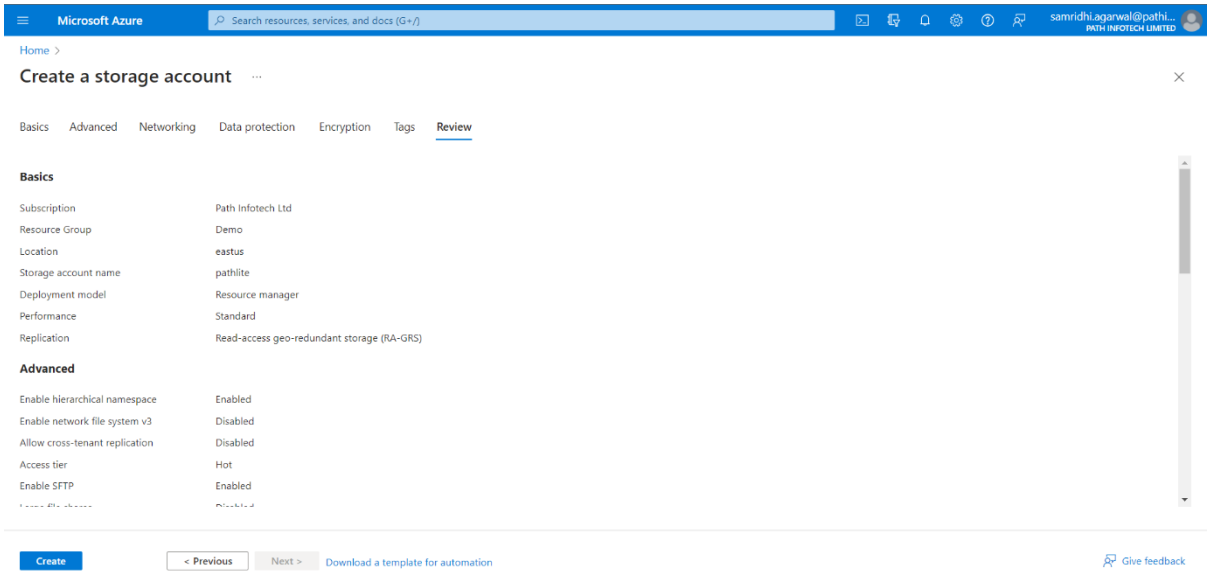


2. Enter random text prompt in the chat session, to judge the outputs without loading data source.



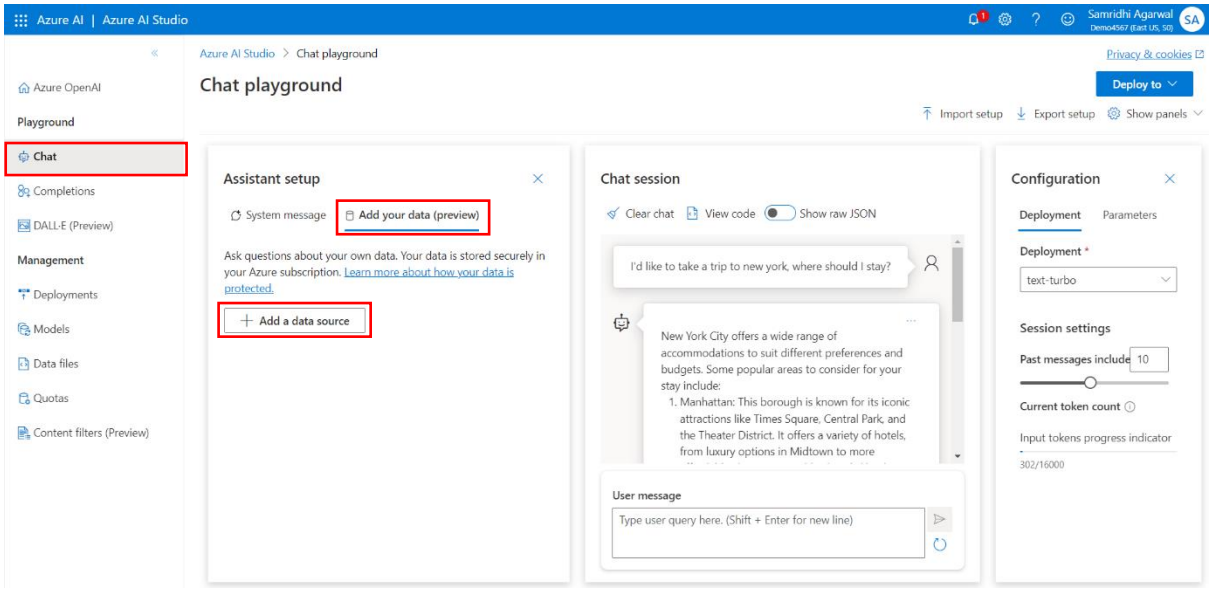
The responses currently are based on the pre-trained broad knowledge that the model has.

3. Create a storage account and a cognitive search resource as well.



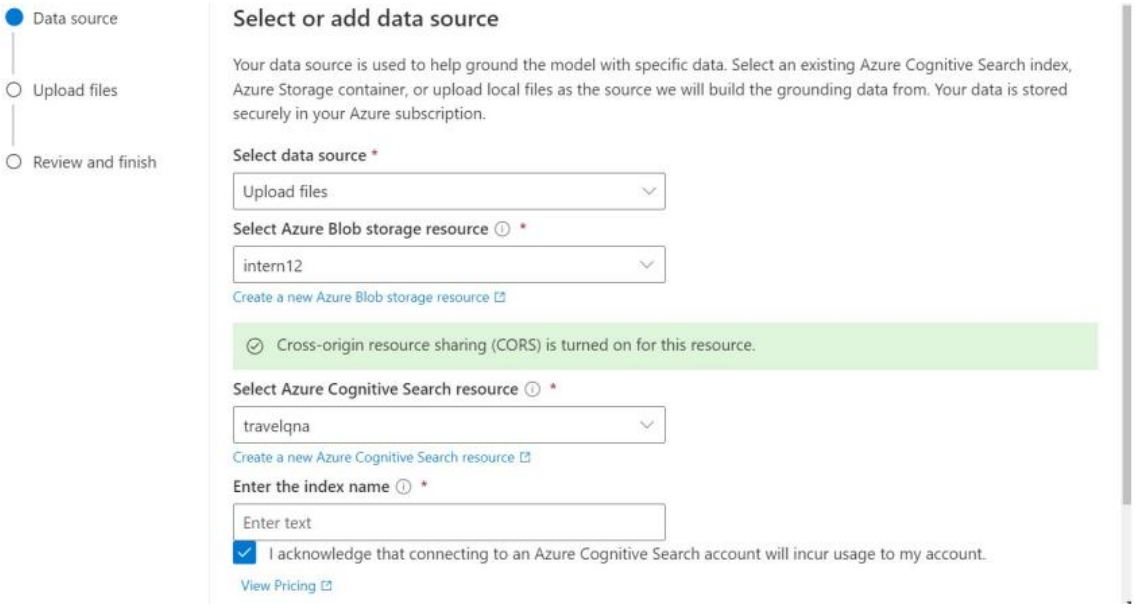
Wait for the deployment to complete then go to the deployed Azure OpenAI resource in Azure portal.

4. Connecting data into chat playground
    a. Navigate to the Chat playground in Azure AI Studio
    b. Select “Add your data” tab in the Assistant Setup pane.
    c. Select “Add a data source” to get data connected.

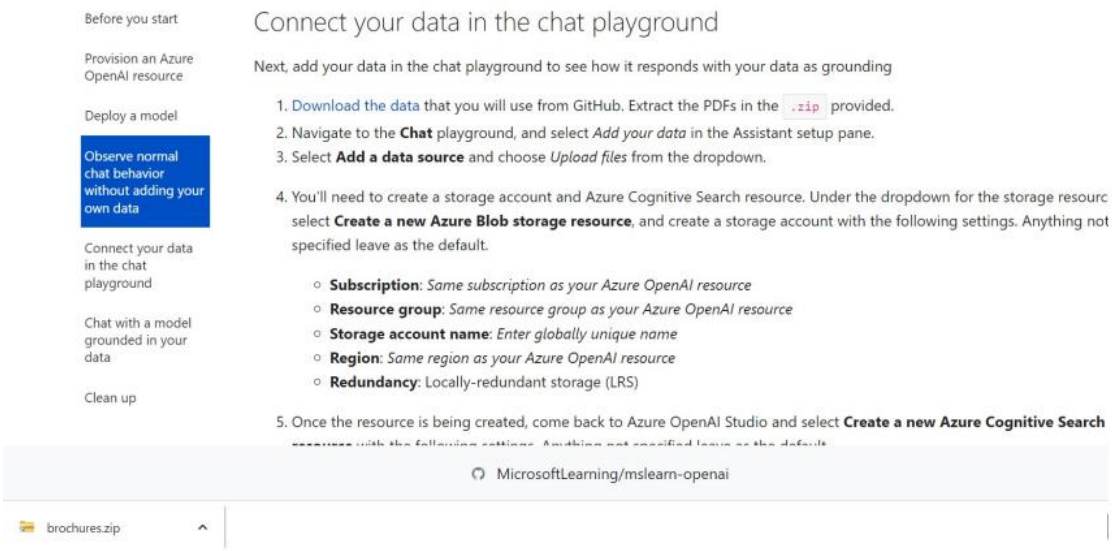


5. In the Add data, enter the following values for data source:

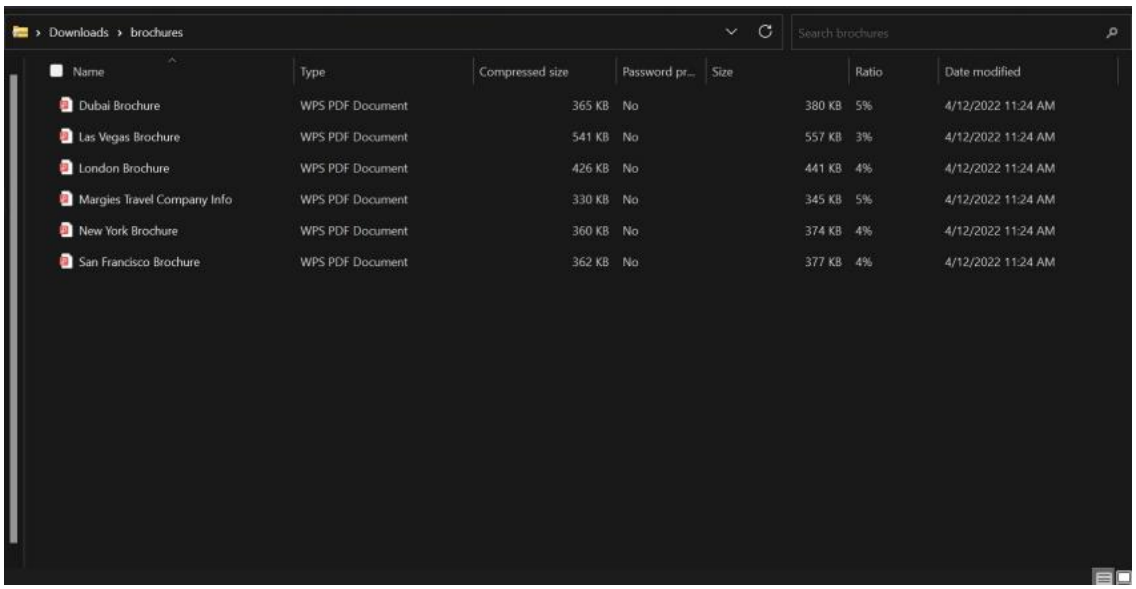
    Data source: Upload files
    Azure Blob Storage Resource: (resource created)
    CORS: “ON”
    Azure Cognitive Search Resource: (resource created)
    Index name: margiestravel



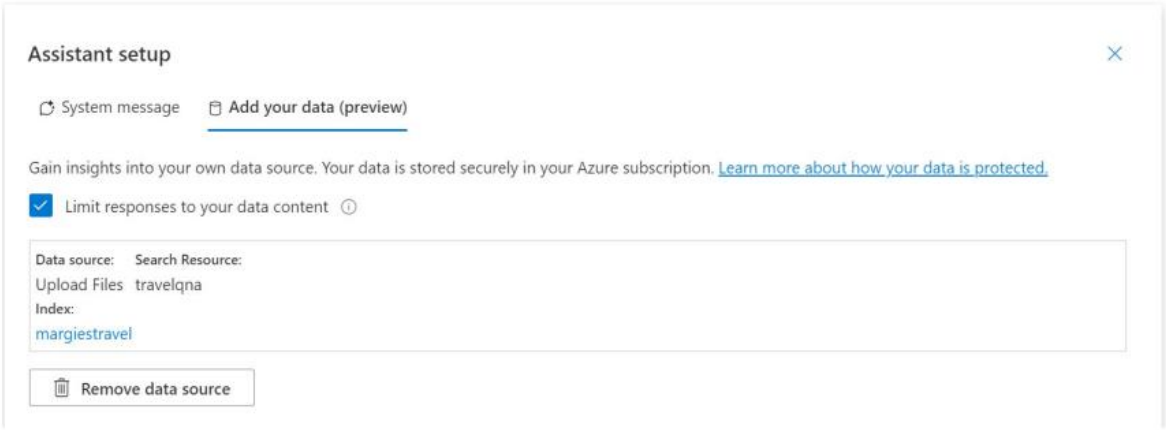
6. Download the files which will be used as the personalized data.



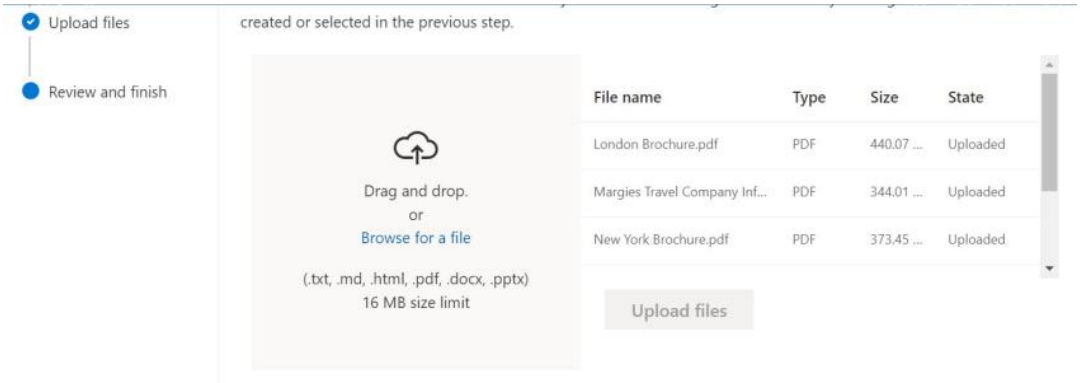
The downloaded zip file contains all the necessary pdf files to be used as personalized data.



7. On the upload files page, upload the pdf files downloaded from the Microsoft learning page which had a hyperlink of github which directly got us downloaded sample pdf files which will be treated as personalised data.

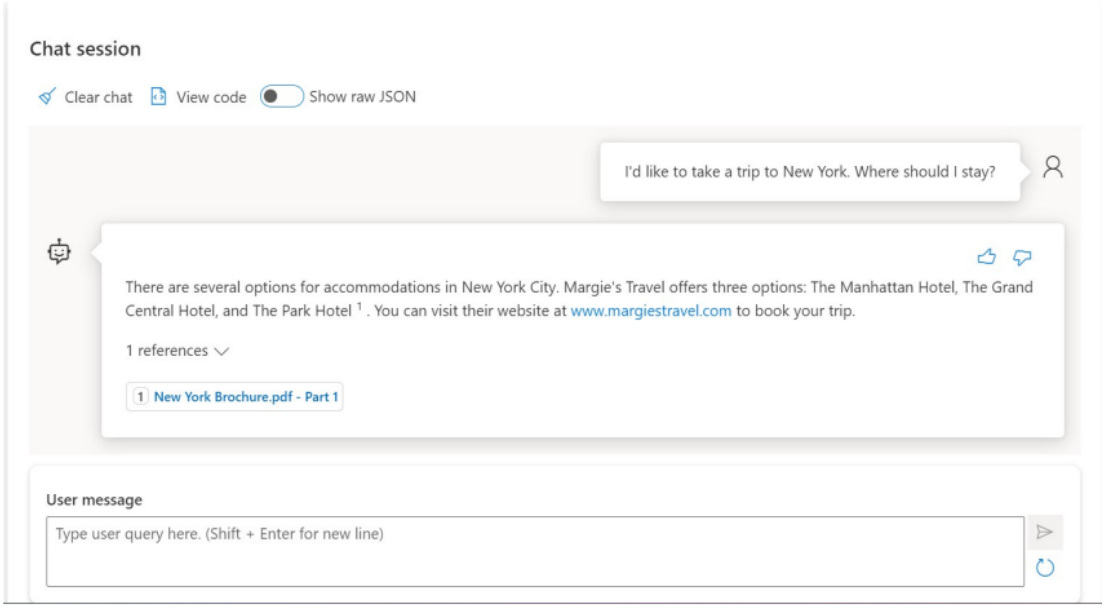


8. The chatbot is ready to be tested with personalized data.



Testing

Checking the chatbot for the personalized data.



The same question has been asked after providing personalized data and gives the answers based on the provided pdfs of data.

Summary

In this module, we saw how Azure OpenAI on personalized data allows developers to use supported AI chat models to reference specific data sources. Connecting personalized data allows the model to reference the specific information provided and its pretrained knowledge to provide more effective responses.

Reference

https://microsoftlearning.github.io/mslearn-openai/instructions/Labs/06-use-own-data.html