

ChatGPT + Enterprise data with Azure OpenAI and Cognitive Search



GITHUB CODESPACES

OPEN



REMOTE - CONTAINERS

OPEN

This sample demonstrates a few approaches for creating ChatGPT-like experiences over your own data using the Retrieval Augmented Generation pattern. It uses Azure OpenAI Service to access the ChatGPT model (gpt-35-turbo), and Azure Cognitive Search for data indexing and retrieval.

The repo includes sample data so it's ready to try end to end. In this sample application we use a fictitious company called Contoso Electronics, and the experience allows its employees to ask questions about the benefits, internal policies, as well as job descriptions and roles.



RAG Architecture

Features

- Chat and Q&A interfaces
- Explores various options to help users evaluate the trustworthiness of responses with citations, tracking of source content, etc.
- Shows possible approaches for data preparation, prompt construction, and orchestration of interaction between model (ChatGPT) and retriever (Cognitive Search)
- Settings directly in the UX to tweak the behavior and experiment with options



Chat screen

Getting Started

IMPORTANT: In order to deploy and run this example, you'll need an **Azure subscription with access enabled for the Azure OpenAI service**. You can request access [here](#). You can also visit [here](#) to get some free Azure credits to get you started.

AZURE RESOURCE COSTS by default this sample will create Azure App Service and Azure Cognitive Search resources that have a monthly cost, as well as Form Recognizer resource that has cost per document page. You can switch them to free versions of each of them if you want to avoid this cost by changing the parameters file under the infra folder (though there are some limits to consider; for example, you can have up to 1 free Cognitive Search resource per subscription, and the free Form Recognizer resource only analyzes the first 2 pages of each document.)

Prerequisites

To Run Locally

- [Azure Developer CLI](#)
- [Python 3+](#)
 - **Important:** Python and the pip package manager must be in the path in Windows for the setup scripts to work.

- **Important:** Ensure you can run `python --version` from console. On Ubuntu, you might need to run `sudo apt install python-is-python3` to link `python` to `python3`.
- [Node.js](#)
- [Git](#)
- [Powershell 7+ \(pwsh\)](#) - For Windows users only.
 - **Important:** Ensure you can run `pwsh.exe` from a PowerShell command. If this fails, you likely need to upgrade PowerShell.

NOTE: Your Azure Account must have `Microsoft.Authorization/roleAssignments/write` permissions, such as [User Access Administrator](#) or [Owner](#).

Note: Rest of document omitted for clarity purposes