Azure OpenAI Workbench User Guide



**Prepared for**

SK C&C

27-Mar-24

Version 0.1 Draft

**Prepared by**

**Han Kyo Jung**

**Contributors**

Table of Contents

[1. Introduction 3](#_Toc162384048)

[2 Azure OpenAI Workbench 개요 4](#_Toc162384049)

[3 GPT Smart Search 아키텍처 5](#_Toc162384050)

[3.1 서비스 흐름 5](#_Toc162384051)

[3.2 서비스 구성 6](#_Toc162384052)

[4 Azure Service배포 및 구성 8](#_Toc162384053)

[4.1 개발 환경 구성하기 8](#_Toc162384054)

[4.1.1 개발자 환경 8](#_Toc162384055)

[4.1.2 Jupyter Lab 환경 9](#_Toc162384056)

[4.2 Azure Service 배포 및 구성 지침 9](#_Toc162384057)

[4.2.1 구성 절차 9](#_Toc162384058)

[4.2.2 Azure Service 배포 10](#_Toc162384059)

[4.2.3 Azure Service 구성 12](#_Toc162384060)

[4.2.4 GPT Smart Search 배포 14](#_Toc162384061)

# Introduction

이 문서는 Azure OpenAI 도입을 검토하는 기업의 Quick Win 과제를 구현하고 검증하는 절차를 표준화한 개발 플랫폼인 Azure OpenAI Workbench에 대해 설명합니다. 이 문서의 주요 목적은 Azure OpenAI Workbench의 아키텍처와 OOO의 흐름을 이해하고, 필요한 요소를 테스트 환경에 구성하는 절차를 이해하는데 있습니다.

따라서, 이 문서에서는 아키텍처, OOO 흐름 및 상세한 구성 지침을 제공합니다.

이 문서에 포함되는 내용은 다음과 같습니다.

* 서비스 개요
* 서비스 구성 및 흐름
* 서비스 배포 및 구성 지침

이 문서는 Azure 및 OpenAI에 대한 기본 지식이 있는 것으로 간주하며, Azure 및 OpenAI에 대한 기본 사항은 포함하지 않습니다. 자세한 내용은 [Azure documentation](https://learn.microsoft.com/en-us/azure/?product=popular)을 참조하세요.

# Azure OpenAI Workbench 개요

대부분의 기업은 다양한 유형의 데이터를 이해할 수 있는 검색엔진과 대화형 Chat Bot을 필요로 합니다. 또한 대화형 Chat Bot은 출처와 함께 문의에 대한 답변을 제공할 수 있어야 합니다. 즉, 비즈니스 데이터에 기반하여 질문을 해석, 이해 및 답변할 수 있는 조직을 위한 비공개 보안 ChatGPT를 원합니다.

Azure OpenAI Workbench의 목표는 고객 환경에서 고객의 데이터를 사용하여 Azure 서비스로 구성된 GPT Smart Search의 구성을 손쉽게 하는 것입니다. 결과물은 다음과 같습니다.

* Bot Framework을 기반으로 구축되어 여러 채널(Web Chat, MS Teams, SMS, E-mail, Slack 등)에 연동할 수 있는 Backend Bot API
* 검색 및 Web Chat UI를 포함한 Frontend Web Application

Azure OpenAI Workbench는 Git 저장소를 통해 제공되며, OpenAI 기반 GPT Smart Search를 구축하는 단계별 지침을 제공하기 위해 만들어졌습니다. 따라서, 각 Notebook은 GPT Smart Search 구성에 필요한 요소들을 빌드하며, Backend Bot API, Frontend Web Application을 빌드하는 것으로 끝납니다.

# GPT Smart Search 아키텍처

고객의 데이터를 사용하여 Azure 서비스로 구성되는 GPT Smart Search의 High-level 아키텍처는 다음과 같습니다.



Figure . GPT Smart Search High-level아키텍처

## 서비스 흐름

1. 사용자는 Frontend Web Application의 WebChat UI를 통해 질문을 합니다.
2. Backend Bot API는 GPT-3.5 or GPT-4.0 LLM을 사용하여 사용자 질문에 따라 사용할 소스를 결정합니다.
3. 두가지 유형의 소스를 사용할 수 있습니다.
   1. Azure AI Search – Blob Storage의 AI-enriched 문서를 포함합니다.
      1. Employee Handbook of Contoso Electronics
      2. 5권의 PDF 서적, 기술 문서
4. Backend Bot API는 소스에서 사용자 질문에 대한 결과를 검색하고 답변을 생성합니다.
5. 질문 과 답변은 추가 분석을 위해 메모리 역할을 하는 Azure Cosmos DB에 저장됩니다.
6. 질문에 대한 답변이 사용자에게 전달됩니다.

## 서비스 구성

GPT Smart Search는 다음의 서비스들로 구성되어 있습니다.

Table . GPT Smart Search 구성

|  |  |  |
| --- | --- | --- |
| Type | Name | Description |
| Azure Service | Azure Bot | Bot Framework을 사용하여 개발된 Back-end Bot API를 연결하고 이를 MS Teams를 포함한 여러 채널에 노출합니다. |
|  | App Service | Back-end Bot API를 호스팅 합니다. |
|  | Search service | Azure AI Service와 Azure OpenAI의 text-embedding-ada-002 모델을 활용하여 구조화되지 않은 문서를 인덱싱하고, 하이브리드 검색(텍스트 및 Vector 검색 결합) 기능을 통해 의미론적 답변을 제공합니다. |
|  | Azure AI Service | 이미지에 대한 OCR, Document Chunking 등과 같은 Search service의 인덱싱 작업에 필요한 기능을 제공합니다. |
|  | Azure OpenAI Service | GPT-4, GPT-4 Turbo with Vision, GPT-3.5-Turbo 및 Embeddings 모델 시리즈를 포함한 OpenAI의 언어 모델에 대한 REST API 액세스를 제공합니다. |
|  | Azure Storage Account | 고객의 문서를 관리하는 기능을 제공합니다. |
|  | Azure Cosmos DB | 사용자 대화를 저장합니다. |

# Azure Service배포 및 구성

## 개발 환경 구성하기

Search Service의 구성 및 Backend Bot API의 Function 구현 및 테스트를 위한 개발 환경으로 Jupyter Lab을 사용합니다. IDE 협업 및 표준화를 위해 Visual Studio Code와 Jupyter Lab이 포함된 Azure Machine Learning의 Compute Instance 사용을 권장합니다.

## 개발자 환경

Azure Machine Learning의 Compute 리소스를 사용하여 로컬에서 개발하기 위해 다음 필수 구성 요소가 있어야 합니다.

* [Visual Studio Code](https://code.visualstudio.com/docs/setup/setup-overview)
* [Azure Machine Learning - Remote Extension](https://marketplace.visualstudio.com/items?itemName=ms-toolsai.vscode-ai-remote), [확장 프로그램 설치하기](https://learn.microsoft.com/en-us/azure/machine-learning/how-to-setup-vs-code?view=azureml-api-2#install-the-extension)

Note:

이 기능은 현재 Preview로 제공됩니다. 이 Preview 버전은 서비스 수준 계약 없이 제공되며 프로덕션 워크로드에는 권장되지 않습니다. 특정 기능은 지원되지 않거나 기능이 제한될 수 있습니다.

* [Python](https://www.python.org/downloads/)
* Azure CLI 버전 2.4 이상, “az –version”을 실행하여 Azure CLI 버전이 2.4 이상인지 확인합니다. 최신 버전이 아닌 경우 [사용 중인 운영 체제 또는 플랫폼의 설치 가이드](https://learn.microsoft.com/ko-kr/cli/azure/install-azure-cli)에 따라 설치를 업데이트합니다.

또한 사용자가 외부에서 다운받은 스크립트를 의도치 않게 무분별하게 실행하는 것을 방지하기 위한 Windows 컴퓨터에 대한 PowerShell 실행 정책을 다음 명령을 실행하여 변경합니다.

|  |
| --- |
| PowerShell |
| Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser - |

## Jupyter Lab 환경

클라우드에서 완전 구성 및 관리형 개발 환경으로 사용할 수 있는 Compute Instance를 제공하는 Azure Machine Learning Workspace를 사용합니다. 다음 절차를 완료합니다.

1. [Azure Machine Learning Workspace 만들기](https://learn.microsoft.com/en-us/azure/machine-learning/quickstart-create-resources?view=azureml-api-2#create-the-workspace)
2. [Compute Instance 만들기](https://learn.microsoft.com/ko-kr/azure/machine-learning/how-to-create-compute-instance?view=azureml-api-2&tabs=azure-studio#create)

## Azure Service 배포 및 구성 지침

## 구성 절차

## Azure Service 배포

Custom Template을 활용하여 Notebooks(Azure AI Search, Cognitive Services 등)를 실행하는 데 필요한 모든 Azure 리소스를 생성합니다.

**사전 준비 사항:**

* 모든 Azure 리소스가 포함될 [리소스 그룹을 생성](https://learn.microsoft.com/en-us/azure/azure-resource-manager/management/manage-resource-groups-portal#create-resource-groups)합니다. Azure OpenAI는 다른 리소스 그룹 또는 다른 구독에 있을 수 있습니다.
* Azure OpenAI는 기존 리소스를 사용하거나, 수동으로 신규 생성하는 것을 원칙으로 합니다. 기존 리소스를 사용하지 않는 경우, Azure Portal에서 [수동으로 Azure OpenAI리소스를 생성](https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/create-resource?pivots=web-portal#create-a-resource) 합니다.

|  |  |  |
| --- | --- | --- |
| Field | Value | Description |
| Region | Sweden Central | 다음 모델이 지원되는 Region을 선택합니다.   * gpt-35-turbo-1106(or newer) * gpt-4-turbo-1106(or newer) * text-embedding-ada-002(or newer) |
| Name | oai-workbench-{숫자, 영문으로 구성된13자리} | 예, oai-workbench- 2znp775rdhyvo |
| Pricing Tier | Standard S0 |  |

* [Azure OpenAI Studio에서 다음 모델을 배포](https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/create-resource?pivots=web-portal#deploy-a-model)합니다.
  1. gpt-35-turbo-1106(or newer)
  2. gpt-4-turbo-1106(or newer)
  3. text-embedding-ada-002(or newer)
* Azure AI Services Multi-Service account를 이전에 생성한 적이 없는 경우, Azure Portal에서 [Azure AI Services Multi-Service account를 수동으로 생성](https://learn.microsoft.com/en-us/azure/ai-services/multi-service-resource?tabs=windows&pivots=azportal#create-a-new-multi-service-resource)하여 Responsible AI 약관을 읽고 동의합니다. 배포 완료된 리소스는 삭제합니다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

check

Note:

표준 Azure 인프라 구성에서 특정 서비스를 제외하거나 신규로 서비스 추가 필요한 경우, Git Repository 루트 디렉토리의 “azuredeploy.json”을 수정하여 요구 사항을 충족하는 Azure 인프라를 배포 할 수 있습니다.

[](https://portal.azure.com/#create/Microsoft.Template/uri/https%3A%2F%2Fraw.githubusercontent.com%2Fendingone%2FAzure-AI-Search-Azure-OpenAI-Workbench%2Fmain%2Fazuredeploy.json)

## Azure Service 구성

다음 절차에 따라 서비스 구성 작업을 완료합니다. 해당 작업은 “4.1.2 Jupyter Lab 환경” 섹션에서 생성한 개발 환경과 아래 Git Repository의 Notebook을 사용합니다.

* Git Repository: https://github.com/endingone/Azure-AI-Search-Azure-OpenAI-Workbench

**사전 준비 사항:**

* “4.1.2 Jupyter Lab 환경”에서 생성한 Compute Instance에 Git Repository를 복제합니다.
  1. Machine Learning Studio를 사용하여 해당 Compute Instance의 터미널에 접속합니다.

텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

click

* 1. 터미널에 다음 명령을 실행하여 Git Repository를 복제합니다. 작업 폴더는 사용자 계정의 루트 폴더입니다.

|  |
| --- |
| Bash |
| git clone git@github.com:endingone/Azure-AI-Search-Azure-OpenAI-Workbench.git |

* 1. 터미널에서 다음 명령을 실행하여 Notebook 실행에 종속된 Python 패키지를 설치합니다.

|  |
| --- |
| Bash |
| conda activate azureml\_py310\_sdkv2  pip install -r ./common/requirements.txt |

Note:

저장소가 비공개인 경우 [SSH를 사용한 Git 계정 인증 설정](https://learn.microsoft.com/en-us/azure/machine-learning/concept-train-model-git-integration?view=azureml-api-2&tabs=python#authenticate-your-git-account-with-ssh)이 필요합니다.

* Desktop의 Visual Studio Code를 사용하여 원격 Compute Instance의 파일 관리하고 Notebook을 실행할 수 있도록 [Machine Learning Studio의 Notebooks 또는 Compute 섹션에서 해당 Compute Instance에 대한 연결](https://learn.microsoft.com/en-us/azure/machine-learning/how-to-launch-vs-code-remote?view=azureml-api-2&tabs=vscode-desktop#use-vs-code-as-your-workspace-ide)을 만듭니다.
* “4.2.2 Azure Service 배포”에서 생성된 서비스의 고유한 값으로 credentials.env 파일을 편집합니다.

노트북을 실행하여 Azure Service를 구성합니다.

Note:

노트북은 상호 종속성을 가지고 있습니다. 순차적으로 실행하여야 합니다.

#Notebook 1: Create an Index with Built-In Indexer in Azure Search Service

#Notebook 2: Searching with Azure OpenAI ChatGPT model

#Notebook 3: Create an index with Custom Indexer implemented by User

#Notebook 4: Implementing Memory to Power Conversations

#Notebook 5: Building and Testing RAG Bot

#Notebook 6: All Together - GPT Smart Search

## GPT Smart Search 배포

다음의 각 지침에 따라 Backend Bot API와 Frontend Web Application을 배포합니다.

* [Backend Bot AP 배포하기](https://github.com/endingone/Azure-AI-Search-Azure-OpenAI-Workbench/blob/main/apps/backend/README.md)
* [Frontend Web Application 배포하기](https://github.com/endingone/Azure-AI-Search-Azure-OpenAI-Workbench/blob/main/apps/frontend/README.md)