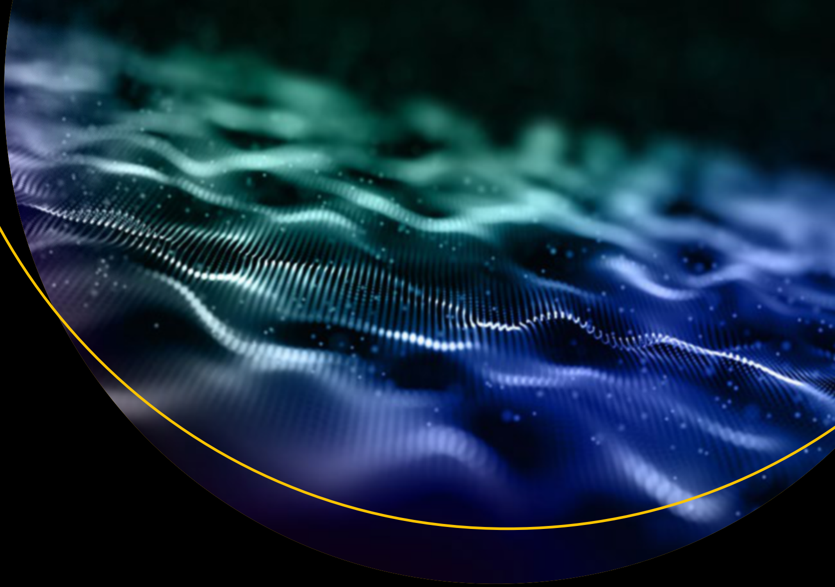

Getting Started with Azure OpenAI

Deploying and Managing Azure AI and Azure OpenAI Solutions

Shimon Ifrah

Apress®

Getting Started with Azure OpenAI

Deploying and Managing Azure AI and Azure OpenAI Solutions

Shimon Ifrah

Apress®

Getting Started with Azure OpenAI: Deploying and Managing Azure AI and Azure OpenAI Solutions

Shimon Ifrah
Melbourne, VIC, Australia

ISBN-13 (pbk): 979-8-8688-0598-1 ISBN-13 (electronic): 979-8-8688-0599-8
https://doi.org/10.1007/979-8-8688-0599-8

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Smriti Srivastava
Development Editor: Laura Berendson
Editorial Assistant: Kripa Joseph

Cover designed by eStudioCalamar

Cover image designed by Freepik (www.freepik.com)

Distributed to the book trade worldwide by Springer Science+Business Media New York, 1 New York Plaza, Suite 4600, New York, NY 10004-1562, USA. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at http://www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub. For more detailed information, please visit https://www.apress.com/gp/services/source-code.

If disposing of this product, please recycle the paper

Table of Contents

About the Author

Shimon Ifrah is a solution architect, writer, tech blogger, and an author with over 15 years of experience in the design, management, and deployment of information technology systems, applications, and networks. In the last decade, Shimon has specialized in cloud computing and containerized applications on Microsoft Azure, Azure AI, Microsoft 365, Azure DevOps, and .NET. Shimon also holds over 20 vendor certificates from Microsoft, AWS, VMware, Oracle, and Cisco. During his career in the IT industry, he has worked for some of the world's largest managed services and technology companies, assisting them in designing and managing systems used by millions of people every day. He is based in Melbourne, Australia.

About the Technical Reviewer

Kasam Shaikh is a prominent figure in India's artificial intelligence landscape, holding the distinction of being one of the country's first four Microsoft Most Valuable Professionals (MVPs) in AI. Currently serving as a senior architect, Kasam boasts an impressive track record as an author, having authored five best-selling books dedicated to Azure and AI technologies. Beyond his writing endeavors, Kasam is recognized as a Microsoft Certified Trainer (MCT) and influential tech YouTuber (@mekasamshaikh). He also leads the largest online Azure AI community, known as DearAzure | Azure INDIA, and is a globally renowned AI speaker. His commitment to knowledge sharing extends to contributions to Microsoft Learn, where he plays a pivotal role.

Within the realm of AI, Kasam is a respected subject matter expert (SME) in generative AI for the cloud, complementing his role as a senior cloud architect. He actively promotes the adoption of No Code and Azure OpenAI solutions and possesses a strong foundation in hybrid and cross-cloud practices. Kasam Shaikh's versatility and expertise make him an invaluable asset in the rapidly evolving landscape of technology, contributing significantly to the advancement of Azure and AI.

In summary, Kasam Shaikh is a multifaceted professional who excels in both technical expertise and knowledge dissemination. His contributions span writing, training, community leadership, public speaking, and architecture, establishing him as a true luminary in the world of Azure and AI. Kasam was recently recognized as the top voice in AI by LinkedIn, making him the sole exclusive Indian professional acknowledged by both Microsoft and LinkedIn for his contributions to the world of artificial intelligence!

CHAPTER 1

Introduction to Azure AI and OpenAI

Thank you for choosing this book. *Getting Started with Azure OpenAI* is my fifth book and third book about Microsoft Azure. In my previous two Azure books, I focused on Microsoft Azure container services (AKS, ACR, Docker, etc.) and how to use infrastructure-as-code tools (Terraform) to deploy services to Azure.

As you embark on this learning journey, this book will guide you through the process of deploying and developing generative artificial intelligence (Gen-AI) solutions using Azure OpenAI services on Microsoft Azure. Your active participation in this process is key to your success, and therefore, I packed this book with many hands-on labs to help you get through the learning process.

In this book, I will guide you on how to develop Gen-AI solutions using the Azure OpenAI platform, with a step-by-step approach to setting up the services from the ground up and deploying them to Microsoft Azure. Besides using the Azure SDK for .NET, we will also utilize the Postman API client to access Azure OpenAI services like Whisper (text to speech), DALL-E (image generation), and the latest GPT-4o model.

Before we get started with Azure OpenAI, let's focus on Azure SDK for .NET. SDK stands for software development kit, a collection of tools and libraries that help developers create applications for a specific platform or service (Azure).

S. Ifrah, *Getting Started with Azure OpenAI*, https://doi.org/10.1007/979-8-8688-0599-8_1

In most cases, an SDK contains an application programming interface (API) that defines how the developer can interact with a platform (Azure) using tools (.NET, C#, VC Code) that facilitate development and deployment.

In this book, we will use the term Azure SDKs to refer to the libraries that Microsoft provides for various Azure services, such as Azure Machine Learning, Azure Cognitive Services, Azure Bot Service, etc. The Azure SDKs are also available in different programming languages like Python, C#, Java, JavaScript, and more.

The Azure SDKs enable you to access the features and capabilities of Azure services from your code without having to deal with low-level details or complex protocols. With the Azure OpenAI SDK library, we can create, configure, train, deploy, and manage AI models on Azure and integrate them with other Azure services and applications.

In this chapter, I will focus on the theoretical part of Azure OpenAI and provide a lot of background that will help you understand the practical part of the book. In this chapter, you will understand the following concepts:

- Models
- Prompt engineering
- Service limits
- Capacity limits
- Cost of running an AI model
- Tokens
- Model servicing

About Azure and Azure OpenAI

Azure is Microsoft's cloud computing platform that offers a wide range of cloud services and solutions for developers and organizations of all sizes. Azure AI is a set of tools and services built into Azure that enable developers to build Gen-AI-based applications using the open source framework of OpenAI and deploy them in Azure.

OpenAI services deployed in Azure run on Azure's robust security and compliance infrastructure and enjoy the high availability and redundancy of Azure datacenter infrastructure that includes 60 regions and 160 datacenters worldwide.

OpenAI is a research and nonprofit organization that aims to create and promote friendly AI that can benefit humanity.

This book is designed to help you get started with Azure OpenAI and learn how to use its features and capabilities using the Azure SDK for .NET to create and deploy AI models for different scenarios.

In the book, we will use multiple programming tools like Azure CLI and Azure PowerShell to deploy the underlying infrastructure services Azure OpenAI uses.

By the end of this book, you will have a solid understanding of Azure OpenAI and the technical knowledge to set up the underlying Azure infrastructure needed for Azure OpenAI and Azure OpenAI services.

Azure AI vs. Azure OpenAI

Before we get started, we must first understand the big difference between the following two services:

- Azure AI
- Azure OpenAI

Azure AI Services

Azure AI Services, formally known as Cognitive Services, was first introduced in 2015 as a set of cloud-based services for developers and data scientists to build intelligent applications using Microsoft's artificial intelligence (AI) and machine learning (ML) capabilities.

Azure AI was launched in 2023 as a rebrand for the existing Cognitive Services suite, including the famous OpenAI.

OpenAI is a research organization founded in 2015 by a group of prominent entrepreneurs, investors, and scientists. The mission of OpenAI is to ensure that artificial intelligence (AI) can be developed in a safe and beneficial way for humanity without being constrained by profit motives or corporate agendas.

OpenAI's most notable projects and capabilities include

- ChatGPT – Large-scale language model (LLM) that generates text and engages in conversations
- DALL-E – LLM model that generates images from text descriptions
- Codex – LLM model that understands and generates code and what powers the GitHub Copilot service
- Whisper – LLM model for speech recognition that can transcribe and translate language to text

On top of the OpenAI Services, Azure AI Services also offers the following core AI Services:

- Azure AI Search – AI service that enables developers to add search capabilities to their applications
- Azure OpenAI – As discussed, a set of tools and services built into Azure that enable developers to use the open source framework OpenAI and deploy them in Azure.

- Bot Service – Enables developers to build, connect, test, and deploy intelligent bots
- Content Safety – AI service that helps detect and filter out potentially unsafe content
- Custom Vision – AI service that enables developers to build custom image classification models
- Document Vision – AI service that enables developers to extract information from documents
- Document Intelligence – AI service that enables developers to extract insights and information from documents
- Face – an AI service that enables developers to detect and analyze faces in images
- Language – an AI service that enables developers to process natural language text
- Speech – an AI service that enables developers to convert speech to text and vice versa
- Translator – an AI service that enables developers to translate text between languages
- Vision – an AI service that enables developers to analyze and understand images

We can use the Azure AI services using the following .NET packages available on NuGet.

In case you are new to .NET, Nuget is a package manager for .NET development, and it allows developers to install, update, and manage libraries and dependencies for their projects.

Packages are installed using the Dotnet CLI, PowerShell, VS Code, and package references. For example, we use the following command to install a NuGet package using the Dotnet CLI.

```
dotnet add package Azure.AI.OpenAI --version 1.0.0-beta.16
```

Nuget holds over 200,000 packages that can be accessed using Visual Studio, the dotnet CLI, or the Nuget website.

Table 1-1 shows the Azure AI packages.

Table 1-1. *Services*

Service Name	.NET Package Details
Azure AI Search	Azure.Search.Documents
Azure OpenAI	Azure.AI.OpenAI
Bot Service	Azure.ResourceManager.BotService
Content Safety	Azure.AI.ContentSafety
Custom Vision	Microsoft.Azure.CognitiveServices.Vision.CustomVision.Prediction Microsoft.Azure.CognitiveServices.Vision.CustomVision.Training
Document Vision	Azure.AI.Vision.Core
Document Intelligence	Azure.AI.DocumentIntelligence
Face	Microsoft.Azure.CognitiveServices.Vision.Face
Language	Azure.AI.TextAnalytics
Speech	Microsoft.CognitiveServices.Speech
Translator	Azure.AI.Translation.Document
Vision	Microsoft.Azure.CognitiveServices.Vision.ComputerVision

Azure OpenAI

Now that we know about the capabilities of Azure AI Services, it is time we understand the capabilities of Azure OpenAI. As explained in the previous section, Azure OpenAI is one of the AI services Azure AI offers.

The reason Azure OpenAI receives so much attention is that it offers OpenAI services under the Microsoft Azure umbrella and allows large organizations to take advantage of existing investments in Azure and develop OpenAI services without needing to make too many changes to their infrastructure or security and compliance policies.

Azure OpenAI offers Azure customers and .NET developers the option to use existing tools, libraries, and code to develop the most advanced AI capabilities developed by OpenAI.

The most common use case for Azure OpenAI is the Azure OpenAI On Your Data, where customers can use OpenAI capabilities on their data, which range from databases to documents and more.

Azure OpenAI also offers infrastructure capabilities like private networking, AI content filtering, and high availability of data centers.

Understanding Prompt Engineering and GPT Models

Prompt engineering is one of the most important concepts of working with OpenAI and other LLMs. Prompt engineering is writing effective inputs for language models like GPT-4 that can perform various natural language tasks.

An input prompt consists of a query that specifies the task and provides some context and a response, which is the model's output based on the query.

GPT-4 is a large-scale language model that uses deep neural networks to learn from a massive amount of data and generate natural language responses as output.

Large Language Models (LLMs)

LLMs are a category of models designed to understand and generate human language. They are characterized by their large number of parameters, which enable them to capture complex linguistic patterns and knowledge.

GPT Models

GPT models are a specific type of LLM developed by OpenAI. They are based on the Transformer architecture and are designed primarily for generative tasks, meaning they can generate coherent and contextually relevant text.

Key Differences Between LLMs and GPT

Specificity: All GPT models are LLMs, but not all LLMs are GPT models. GPT is a specific implementation within the broader category of LLMs.

Architecture: While GPT models use the Transformer architecture, other LLMs might use different variations or optimizations of the Transformer model.

Training Objectives: GPT models are generally trained with a focus on generative tasks, whereas other LLMs might be optimized for different objectives, such as bidirectional understanding in the case of BERT.

In short, while GPT models are a subset of LLMs known for their text generation capabilities, LLMs encompass a wider range of models designed for various NLP tasks.

Prompt Engineering Strategies

- Choose the right format and tone for the query and the response. The more specific the request, the output will be accurate.
- Provide enough context, be specific, and provide examples for the model to understand the task and the desired output.
- Experiment with different variations and combinations of queries and responses to find the right prompt.
- Always evaluate the model's outputs using external sources, metrics, fluency, diversity, and alignment with the task goals.
- Provide feedback regarding the outputs, and correct it to improve the model's learning and trustworthiness.

Prompt engineering is a must skill for working with GPT models and other LLMs because it helps produce the right results from an LLM.

Azure OpenAI Models

Azure OpenAI offers almost the same range of OpenAI models with the latest API version. It is also a good idea to consider each model's cost when working with models, as prices are not the same for all models.

Azure OpenAI offering includes access to the following models:

- GPT-4 and GPT-4 Turbo – The most powerful language models ever created, with billions of parameters and unprecedented speed and accuracy

- GPT-3.5 – A scaled-down version of GPT-4, still capable of generating high-quality natural language for a variety of domains and tasks
- Embeddings – A service that provides vector representations of words, sentences, and documents, enabling semantic similarity and clustering analysis
- DALL-E – A generative model that can create stunning images from natural language descriptions, such as "a cat wearing a bow tie" or "a snail made of harp"
- Whisper – A service that converts text to audio with a choice of voices, languages, and emotions

Using Azure OpenAI models works on the principle of using and accessing any REST API service and makes no difference if we are pointing to an OpenAI API endpoint or an Azure OpenAI API endpoint.

Azure OpenAI gives us access to the latest and newest OpenAI models like GPT-4, GPT-3.5 Turbo, and more. These models allow us to:

- Generate content
- Summarize text
- Use images with prompts
- Use semantic search
- Generate code and scripts

These services are available to us using the Azure OpenAI REST API endpoints or using the Azure SDKs. We can also use these services with Azure OpenAI Studio, which is a portal that allows us to use OpenAI services using a GUI interface. We will cover the OpenAI Studio later in the book.

Limits, Capacity, Context, and Tokens

One of the things we need to be aware of when working with LLMs is their limited capacity to process and generate text. Every LLM uses tokens to calculate words, subwords, or characters that can be combined to form words and sentences. That calculation is also used to calculate the cost of running an LLM.

When working with Azure OpenAI deployment, Azure allocates each subscription a service quota that provides a rate limit per model deployment in order to maintain service reliability.

A rate limit is calculated using the Tokens-Per-Minute (TPM) in multiples of 1000 the deployment consumes. For example, the GPT-4 model has a 40K TPM limit per deployment. On top of the Azure RPM limit, each model has a maximum request limit in tokens. Table 1-2 shows the list of GPT-4 models at the time of writing this book.

Table 1-2. *Max request limit*

Model ID	Max Request (Tokens)
gpt-4 (0314)	8,192 (8K)
gpt-4-32k (0314)	32,768 (32K)
gpt-4 (0613)	8,192 (8K)
gpt-4-32k (0613)	32,768 (32K)
gpt-4 (1106-Preview) GPT-4 Turbo Preview	Input: 128,000 (128K) Output: 4,096 (4K)
gpt-4 (0125-Preview)1GPT-4 Turbo Preview	Input: 128,000 (128K) Output: 4,096 (4K)
gpt-4 (vision-preview)2GPT-4 Turbo with Vision Preview	Input: 128,000 (128K) Output: 4,096 (4K)

Understanding Context and K

When working with models and as shown in Table 1-2, each model has a context limit defined by K. The K next to the number means "Killo" (thousand). When working with the gpt-4-32k model, the 32K means the model can use 32K tokens per context window (also known as session or conversation).

As explained earlier, a token in the context of GPT-4 represents a whole word or a subword. This means the model can generate a text with a maximum of 32K in a single conversation (also known as context window).

A context window refers to the maximum amount of text (in tokens) that the model can accept as input at one time. This also applies to the amount of output the model can return.

In practice, a 32K context window is a lot, giving the model a large amount of information for reference and output. It also allows us to input a very large amount of data to the model and produce an extensive amount of text.

The Benefits of Using a Large Context Window

When working with an 8K (default) and a 32K LLM model, we can easily understand the following main benefits of using a large (K) context window:

- Produce better results by inputting more information into the model and generating more information like results, etc.
- Broader reference – With a large context, the model can "remember" more information and retain more knowledge.

- Relevance – The more information the model has, the greater the relevance and accuracy of information it produces.

As AI continues to evolve and grow, the focus will be on the size of the context of new models in the upcoming years. More context means more capabilities and more processing power.

When working with models in the current environment, it is recommended to update the model you are working with to the latest model release. A new release means more tokens and up-to-date training data.

Training data (up to) in the context of LLM models refers to the dataset used to train the model. In the case of the gpt-4 (0314) model, the training data dataset is up to September 2021, which means the model doesn't know about events that took place after September 2021.

The latest GPT-4 model, gpt-4 (0125-preview), has a training data that goes up to December 2023. It also has a context window of 128K. This is a major improvement compared to the 0314 model and can make a huge difference to the results on an LLM application, not to mention the results the model can output.

Another point to consider is the retirement dates of the module; as of writing these lines, the 0314 model will be retired after July 5, 2024. For that reason, Azure introduced a feature that will automatically update models.

Cost of Azure OpenAI Models

As mentioned earlier, each model has a context limit and also a price, and choosing the right model is an important task. Just like selecting the size of an Azure Virtual Machine, we also need to take the same approach when working with OpenAI models.