✓ 100 XP ▶

Use Azure OpenAI REST API

6 minutes

Azure OpenAI offers a REST API for interacting and generating responses that developers can use to add AI functionality to their applications. This unit covers example usage, input and output from the API.

> ⓘ **Note**
>
> Before interacting with the API, you must create an Azure OpenAI resource in the Azure portal, deploy a model in that resource, and retrieve your endpoint and keys. Check out the [Getting started with Azure OpenAI Service](#) to learn how to do that.

For each call to the REST API, you need the endpoint and a key from your Azure OpenAI resource, and the name you gave for your deployed model. In the following examples, the following placeholders are used:

⟦ ⟧ **Expand table**

Placeholder name	Value
`YOUR_ENDPOINT_NAME`	This base endpoint is found in the **Keys & Endpoint** section in the Azure portal. It's the base endpoint of your resource, such as `https://sample.openai.azure.com/`.
`YOUR_API_KEY`	Keys are found in the **Keys & Endpoint** section in the Azure portal. You can use either key for your resource.
`YOUR_DEPLOYMENT_NAME`	This deployment name is the name provided when you deployed your model in the Azure OpenAI Studio.

Chat completions

Once you've deployed a model in your Azure OpenAI resource, you can send a prompt to the service using a `POST` request.

rest

```
curl https://YOUR_ENDPOINT_NAME.openai.azure.com/openai/deployments/YOUR_DE-
PLOYMENT_NAME/chat/completions?api-version=2023-03-15-preview \
  -H "Content-Type: application/json" \
  -H "api-key: YOUR_API_KEY" \
  -d '{"messages":[{"role": "system", "content": "You are a helpful as-
sistant, teaching people about AI."},
{"role": "user", "content": "Does Azure OpenAI support multiple languages?"},
{"role": "assistant", "content": "Yes, Azure OpenAI supports several lan-
guages, and can translate between them."},
{"role": "user", "content": "Do other Azure AI Services support translation
too?"}]}'
```

The response from the API will be similar to the following JSON:

JSON

```
{
    "id": "chatcmpl-6v7mkQj980V1yBec6ETrKPRqFjNw9",
    "object": "chat.completion",
    "created": 1679001781,
    "model": "gpt-35-turbo",
    "usage": {
        "prompt_tokens": 95,
        "completion_tokens": 84,
        "total_tokens": 179
    },
    "choices": [
        {
            "message":
                {
                    "role": "assistant",
                    "content": "Yes, other Azure AI Services also support
translation. Azure AI Services offer translation between multiple languages
for text, documents, or custom translation through Azure AI Services
Translator."
                },
            "finish_reason": "stop",
            "index": 0
        }
    ]
}
```

REST endpoints allow for specifying other optional input parameters, such as `temperature`, `max_tokens` and more. If you'd like to include any of those parameters in your request, add them to the input data with the request.

Embeddings

Embeddings are helpful for specific formats that are easily consumed by machine learning models. To generate embeddings from the input text, `POST` a request to the `embeddings` endpoint.

```rest
curl https://YOUR_ENDPOINT_NAME.openai.azure.com/openai/deployments/YOUR_DE-
PLOYMENT_NAME/embeddings?api-version=2022-12-01 \
  -H "Content-Type: application/json" \
  -H "api-key: YOUR_API_KEY" \
  -d "{\"input\": \"The food was delicious and the waiter...\"}"
```

When generating embeddings, be sure to use a model in Azure OpenAI meant for embeddings. Those models start with `text-embedding` or `text-similarity`, depending on what functionality you're looking for.

The response from the API will be similar to the following JSON:

```JSON
{
  "object": "list",
  "data": [
    {
      "object": "embedding",
      "embedding": [
        0.0172990688066482523,
        -0.0291879814639389515,
        ....
        0.0134544348834753042,
      ],
      "index": 0
    }
  ],
  "model": "text-embedding-ada:002"
}
```

Next unit: Use Azure OpenAI SDK

Continue >