

OpenAI API利用チュートリアル

はじめに

本書は、本サービスが提供するOpenAI API準拠のAPIを利用して一般対話を行うクライアントアプリケーションを作成するチュートリアルです。

本書の対象読者は以下を想定しています。

- 本サービスと連携したシステムや製品開発を行う開発者
- 履歴付き一般対話APIやSearch APIでは実現できない機能を開発したい開発者

なお、本書ではPythonのサンプルコードを記載しておりますが、本サービスのAPIはREST形式のため、他の言語からでもご利用いただけます。

サポートしているAPIとリクエスト・レスポンスの詳細については、リファレンスマニュアル「OpenAI互換API」を参照してください。

- 本ガイドに記載する「<https://<サーバのドメイン名>/>」はGenerative AI FWがインストールされているサーバのドメイン名に置き換えてAPIを実行してください。
- 本サービスではHTTPSに既定では自己証明書を使用しています。そのためAPI利用時に考慮が必要です。詳細は「スタートアップマニュアル（導入準備編）」をご確認ください。

チュートリアルの流れ

本チュートリアルでは、OpenAIライブラリおよびrequestsライブラリを利用した例を記載します。

共通手順

1. (共通) 一般対話API (OpenAI API)を利用するために認証情報を準備する

OpenAIライブラリ

1. OpenAI APIライブラリのインストール
2. Chat completions APIを呼び出すPythonコードを実装し、実行する
3. Chat completions APIでTool callingを行うPythonコードを実装し、実行する

requestsライブラリを利用

1. requestsライブラリのインストール

2. Chat completions APIを呼び出すPythonコードを実装し、実行する
3. Chat completions APIでTool callingを行うPythonコードを実装し、実行する

OpenAI APIライブラリを用いたチュートリアル

OpenAI APIライブラリのインストール

インストールコマンドの例

```
1 pip install openai
```

Chat completions

Pythonコードを実装する

Pythonコードの例

Pythonのエディタを開き、下記のコードを記述します。KEYの値を適切なものに修正し、ファイル名を「openAI_main.py」として保存します。

```
1 from openai import OpenAI
2
3 # KEYはAPIキー
4 # "Bearer "はOpenAIライブラリがセットするので指定しない。
5 KEY = 'abcdefg1234567890'
6 # BASEは本サービスのURL
7 BASE = 'https://<サーバのドメイン名>/genai-oai-api/v1'
8 # MODELはエイリアス名
9 MODEL = 'cotomi-v3.0'
10
11 def func():
12     messages = [ { "role": "system", "content": "あなたはAIアシスタントです" },
13                  { "role": "user", "content": "日本で一番高い山は" } ]
14
15     # インスタンス生成
16     client = OpenAI(
17         api_key = KEY
18     )
19     client.base_url = BASE
20     model = MODEL
21
22     # ストリーム形式
23     response = client.chat.completions.create(
24         model=model,
25         messages=messages,
26         max_tokens=8,
27         stream=True
28     )
29
30     for chunk in response:
31         print(chunk.model_dump_json())
32
33 if __name__ == '__main__':
34     func()
```

Pythonの実行

実行コマンド

```
1 python openAI_main.py
```

実行結果の例

```
1 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "", ...}}
2 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "日本で", ...}}
3 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "一番高い", ...}}
4 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "山は", ...}}
5 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "富士山", ...}}
6 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "です", ...}}
7 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "。", ...}}
8 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "標高", ...}}
9 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "は", ...}}
10 {"id": "chat-33f2c7ab7f534ae4a49e1399a89ef424", "choices": [{"delta": {"content": "", ...}}]
```

Tool calling (Function calling)

Pythonコードを実装する

Pythonコードの例

Pythonのエディタを開き、下記のコードを記述します。KEYの値を適切なものに修正し、ファイル名を「openAI_tool_calling_main.py」として保存します。

```
1 from openai import OpenAI
2
3 # KEYはAPIキー
4 # "Bearer "はOpenAIライブラリがセットするので指定しない。
5 KEY = 'abcdefg1234567890'
6 # BASEは本サービスのURL
7 BASE = 'https://<サーバのドメイン名>/genai-oai-api/v1'
8 # MODELはエイリアス名
9 MODEL = 'cotomi-v3.0'
10
11 def func():
12     messages = [{"role": "user", "content": "名古屋の天気は?"}]
13     tools = [
14         {
15             "type": "function",
16             "function": {
17                 "name": "get_weather",
18                 "description": "指定した地域の現在の天気を取得する",
19                 "parameters": {
20                     "type": "object",
21                     "properties": {
22                         "location": {
23                             "type": "string",
24                             "description": "都道府県と市区町村, 例: '東京都立川市'"
25                         },
26                         "unit": {
27                             "type": "string",
28                             "enum": ["celsius", "fahrenheit"]
29                         }
30                     },
31                     "required": ["location", "unit"]
32                 }
33             }
34         }
35     ]
```

```

34
35     # インスタンス生成
36     client = OpenAI(
37         api_key = KEY
38     )
39     client.base_url = BASE
40     model = MODEL
41
42     # Tool calling
43     response = client.chat.completions.create(
44         model=model,
45         messages=messages,
46         tools=tools,
47         tool_choice="auto"
48     )
49
50     print(response.choices[0].message.tool_calls)
51
52 if __name__ == '__main__':
53     func()

```

Pythonの実行

実行コマンド

```
1 | python openAI_tool_calling_main.py
```

実行結果の例（読みやすいように整形しています）

```

1 | [
2 |   ChatCompletionMessageToolCall(
3 |     id='chatcpl-tool-e271ad2fd26e4b778f198430268a54eb',
4 |     function=Function(
5 |       arguments='{"location": "愛知県名古屋市", "unit": "celsius"}',
6 |       name='get_weather'
7 |     ),
8 |     type='function'
9 |   )
10 | ]

```

requestsライブラリを用いたチュートリアル

requestsライブラリのインストール

インストールコマンドの例

```
1 | pip install requests
```

Chat completions

Pythonコードを実装する

Pythonコードの例

Pythonのエディタを開き、下記のコードを記述します。KEYの値を適切なものに修正し、ファイル名を「openAI_main.py」として保存します。

```
1 import requests
2 import codecs
3
4
5 # KEYはAPIキー
6 KEY = 'abcdefg1234567890'
7 # BASEは本サービスのURL
8 BASE = 'https://<サーバのドメイン名>/genai-oai-api/v1'
9 # MODELはエイリアス名
10 MODEL = 'cotomi-v3.0'
11
12 def func():
13     url = BASE + '/chat/completions'
14     key = 'Bearer ' + KEY
15     payload = { "messages": [ { "role": "system", "content": "あなたはAIアシスタントです" },
16                               { "role": "user", "content": "日本で一番高い山は" } ], "model": MODEL }
17     headers = { 'content-type': 'application/json', 'Authorization': key }
18     # 非ストリーム形式
19     response = requests.post(url, json=payload, headers=headers)
20     print(response.status_code)
21     print(response.text)
22     #print(response.headers)
23
24 if __name__ == '__main__':
25     func()
```

Pythonの実行

実行コマンド

```
1 python openAI_main.py
```

実行結果の例

```
1 200
2 {
3   "id": "chatcmpl-4252e403a4dd47a18bfd8f48639c80b",
4   "object": "chat.completion",
5   "created": 1737082377,
6   "model": "cotomi-v3.0",
7   "choices": [ {
8     "index": 0,
9     "message": {
10      "role": "assistant",
11      "content": "日本で最も高い山は富士山です。...",
12      "tool_calls": [],
13      "logprobs": null,
14      "finish_reason": "stop",
15      "stop_reason": null
16    },
17     "usage": {
18       "prompt_tokens": 16,
19       "total_tokens": 41,
20       "completion_tokens": 25,
21       "prompt_tokens_details": null,
22       "prompt_logprobs": null
23     }
24   }
25 }
```

Tool calling (Function calling)

Pythonコードを実装する

Pythonコードの例

Pythonのエディタを開き、下記のコードを記述します。KEYの値を適切なものに修正し、ファイル名を「openAI_tool_calling_main.py」として保存します。

```
1 import requests
2 import json
3
4 # KEYはAPIキー
5 KEY = 'abcdefg1234567890'
6 # BASEは本サービスのURL
7 BASE = 'https://<サーバのドメイン名>/genai-oai-api/v1'
8 # MODELはエイリアス名
9 MODEL = 'cotomi-v3.0'
10
11 def func():
12     url = BASE + '/chat/completions'
13     key = 'Bearer ' + KEY
14     payload = {
15         "model": MODEL,
16         "messages": [{"role": "user", "content": "名古屋の天気は?"}],
17         "tools": [{
18             "type": "function",
19             "function": {
20                 "name": "get_weather",
21                 "description": "指定した地域の現在の天気を取得する",
22                 "parameters": {
23                     "type": "object",
24                     "properties": {
25                         "location": {
26                             "type": "string",
27                             "description": "都道府県と市区町村, 例: '東京都立川市'"
28                         },
29                         "unit": {
30                             "type": "string",
31                             "enum": ["celsius", "fahrenheit"]
32                         }
33                     },
34                     "required": ["location", "unit"]
35                 }
36             }
37         }],
38         "tool_choice": "auto"
39     }
40     headers = { 'content-type': 'application/json', 'Authorization': key }
41
42     # Tool calling
43     response = requests.post(url, json=payload, headers=headers)
44     print(response.status_code)
45     print(json.dumps(json.loads(response.text), ensure_ascii=False))
46     #print(response.headers)
47
48 if __name__ == '__main__':
49     func()
```

Pythonの実行

実行コマンド

実行結果の例（読みやすいように整形しています）

```
1 200
2 {
3   "choices": [
4     {
5       "finish_reason": "tool_calls",
6       "index": 0,
7       "logprobs": null,
8       "message": {
9         "content": null,
10        "reasoning_content": null,
11        "role": "assistant",
12        "tool_calls": [
13          {
14            "function": {
15              "arguments": "{¥"location¥": ¥"愛知県名古屋市¥", ¥"unit¥": ¥"celsius¥"}",
16              "name": "get_weather"
17            },
18            "id": "chatcmpl-tool-47d833ccf54b474a8a759a2c1c6dfa7a",
19            "type": "function"
20          }
21        ]
22      },
23      "stop_reason": null
24    }
25  ],
26  "created": 1754301685,
27  "id": "chatcmpl-66dd09fd4143958c73fea7fa6dbfa183",
28  "model": "cotomi-v3.0",
29  "object": "chat.completion",
30  "prompt_logprobs": null,
31  "usage": {
32    "completion_tokens": 31,
33    "prompt_tokens": 304,
34    "prompt_tokens_details": null,
35    "total_tokens": 335
36  }
37 }
```