🏠 ■ Modules ■ Data connection ■ Retrievers ■ Integrations ■ ChatGPT Plugin

ChatGPT Plugin

> OpenAI plugins connect ChatGPT to third-party applications. These plugins enable ChatGPT to interact with APIs defined by developers, enhancing ChatGPT's capabilities and allowing it to perform a wide range of actions.

> Plugins can allow ChatGPT to do things like:
>
> - Retrieve real-time information; e.g., sports scores, stock prices, the latest news, etc.
> - Retrieve knowledge-base information; e.g., company docs, personal notes, etc.
> - Perform actions on behalf of the user; e.g., booking a flight, ordering food, etc.

This notebook shows how to use the ChatGPT Retriever Plugin within LangChain.

```
# STEP 1: Load

# Load documents using LangChain's DocumentLoaders
# This is from https://langchain.readthedocs.io/en/latest/modules/document_loaders/examples/csv.html

from langchain.document_loaders.csv_loader import CSVLoader

loader = CSVLoader(
    file_path="../../document_loaders/examples/example_data/mlb_teams_2012.csv"
)
data = loader.load()


# STEP 2: Convert
# Convert Document to format expected by https://github.com/openai/chatgpt-retrieval-plugin
from typing import List
from langchain.docstore.document import Document
import json


def write_json(path: str, documents: List[Document]) -> None:
    results = [{"text": doc.page_content} for doc in documents]
    with open(path, "w") as f:
        json.dump(results, f, indent=2)


write_json("foo.json", data)

# STEP 3: Use

# Ingest this as you would any other json file in https://github.com/openai/chatgpt-retrieval-plugin/tree/main/scripts/process_json
```

Using the ChatGPT Retriever Plugin

Okay, so we've created the ChatGPT Retriever Plugin, but how do we actually use it?

The below code walks through how to do that.

We want to use `ChatGPTPluginRetriever` so we have to get the OpenAI API Key.

```
import os
import getpass

os.environ["OPENAI_API_KEY"] = getpass.getpass("OpenAI API Key:")
```

```
    OpenAI API Key: ········
```

```
from langchain.retrievers import ChatGPTPluginRetriever
```

```
retriever = ChatGPTPluginRetriever(url="http://0.0.0.0:8000", bearer_token="foo")
```

```
retriever.get_relevant_documents("alice's phone number")
```

```
    [Document(page_content="This is Alice's phone number: 123-456-7890", lookup_str='', metadata={'id':
'456_0', 'metadata': {'source': 'email', 'source_id': '567', 'url': None, 'created_at': '1609592400.0',
'author': 'Alice', 'document_id': '456'}, 'embedding': None, 'score': 0.925571561}, lookup_index=0),
     Document(page_content='This is a document about something', lookup_str='', metadata={'id': '123_0',
'metadata': {'source': 'file', 'source_id': 'https://example.com/doc1', 'url': 'https://example.com/doc1',
'created_at': '1609502400.0', 'author': 'Alice', 'document_id': '123'}, 'embedding': None, 'score':
0.6987589}, lookup_index=0),
     Document(page_content='Team: Angels "Payroll (millions)": 154.49 "Wins": 89', lookup_str='', metadata=
{'id': '59c2c0c1-ae3f-4272-a1da-f44a723ea631_0', 'metadata': {'source': None, 'source_id': None, 'url': None,
'created_at': None, 'author': None, 'document_id': '59c2c0c1-ae3f-4272-a1da-f44a723ea631'}, 'embedding':
None, 'score': 0.697888613}, lookup_index=0)]
```