動物趣味事實問答

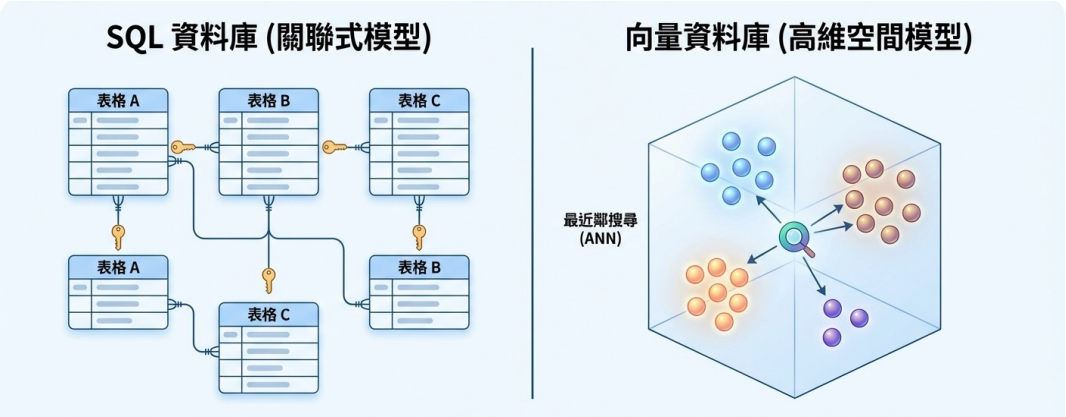
向量資料庫建立與資料搜尋

Ryan Chung

20260108

需求

- 將資料轉換成向量資料庫
- 使用者輸入一句話後，能找到最相近的資料



資料集：Animal Fun Fact

	A	B	C	D	E
1	animal_name	source	text	media_link	wikipedia_link
2	aardvark	https://www.animalfactsencyclopedia.com/Aardvark-facts.html	Aardvarks are sometimes called "ant bears", "earth pigs", and "cape anteaters"		/wiki/Aardvark
3	aardvark	https://www.animalfactsencyclopedia.com/Aardvark-facts.html	Aardvarks have rather primitive brains that are very small for the size of the		/wiki/Aardvark
4	aardvark	https://www.animalfactsencyclopedia.com/Aardvark-facts.html	Aardvarks teeth are lined with fine upright tubes and have no roots or enamel.		/wiki/Aardvark
5	aardvark	https://www.animalfactsencyclopedia.com/Aardvark-facts.html	The aardvarks Latin family name "Tubulidentata" means "tube toothed"		/wiki/Aardvark
6	aardvark	https://www.animalfactsencyclopedia.com/Aardvark-facts.html	Baby aardvarks are born with front teeth that fall out and never grow back.		/wiki/Aardvark
7	aardvark	https://www.animalfactsencyclopedia.com/Aardvark-facts.html	Aardvarks are living fossils not having changed for millions of years.		/wiki/Aardvark
8	aardvark	https://www.animalfactsencyclopedia.com/Aardvark-facts.html	Aardvarks will occasionally stand, and even take a step or two, on their hind legs		/wiki/Aardvark
9	aardvark	https://www.animalfactsencyclopedia.com/Aardvark-facts.html	Aardvarks can use their powerful tails as a whip-like weapon of defense.		/wiki/Aardvark
10	african wild d	https://www.animalfactsencyclopedia.com/African-wild-dog-facts	Wild dogs are known by many different names including painted dog, painted wolf, c		/wiki/African_wild_dog
11	african wild d	https://www.animalfactsencyclopedia.com/African-wild-dog-facts	They are the most efficient hunters of any large predator with an 80% success rate.		/wiki/African_wild_dog

https://github.com/ryanchung403/dataset/blob/main/animal-fun-facts-dataset.csv

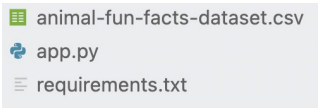
建立專案

- 新增資料夾 animal-fun-facts-search

  app.py

  requirements.txt

  animal-fun-facts-dataset.csv



app.py

```python
from langchain_community.vectorstores import FAISS
import pandas as pd
from langchain_huggingface import HuggingFaceEmbeddings

# Load dataset
animal_data = pd.read_csv("animal-fun-facts-dataset.csv")

embedding_function = HuggingFaceEmbeddings(
    model_name="sentence-transformers/all-MiniLM-L6-v2"
)

metadatas = []
for i, row in animal_data.iterrows():
    metadatas.append(
        {
            "Animal Name": row["animal_name"],
            "Source URL": row["source"],
            # "Media URL": row["media_link"],
            # "Wikipedia URL": row["wikipedia_link"],
        }
    )

animal_data["text"] = animal_data["text"].astype(str)
faiss = FAISS.from_texts(animal_data["text"].to_list(), embedding_function, metadatas)
faiss.similarity_search_with_score("What is ship of the desert?", 3)
```

- 載入模組
- 載入資料集
- 嵌入函數
  - 模型選用
- 準備資料
  - metadata
  - 型態轉換
- 產生向量資料庫
- 資料搜尋

https://gist.github.com/ryanchung403/a4c5390830bbdbbe094dd59999165ae7

requirements.txt

- 安裝需求模組

```
langchain_community
pandas
langchain_huggingface
faiss-cpu
sentence-transformers
ipykernel
```

建立虛擬環境 & 安裝套件

- 在VS Code 中，停在app.py頁面
- 點擊右下角Python版本數字
- 上方選擇「+建立虛擬環境...」
- Venv
- 選擇Python版本，例如「Python 3.13.3」
- 勾選 requirements.txt，按下確定

結果確認

- 成功找到答案！

```
[69]  faiss.similarity_search_with_score("What is ship of the desert?", 3)
      ✓ 0.0s
```

```
[(Document(page_content='The camel is known as the "Ship of the Desert"', metadata={'Animal Name': 'camel', 'Source URL': '
  0.71180713),
 (Document(page_content='Found in the African Sahara Desert!', metadata={'Animal Name': 'fennec fox', 'Source URL': 'https:
  0.9218745),
 (Document(page_content='The chameleon of the seas!', metadata={'Animal Name': 'summer flounder', 'Source URL': 'https://a-
  0.9567355)]
```

向量資料庫的匯出與匯入

- 匯出(寫在原本的app.py最下方)

```
#export the vector store to disk
faiss.save_local("faiss_db","index")
```

- 匯入(寫在要用到這個向量資料庫的py檔)

```
# import the vector-db from disk
from langchain_community.vectorstores import FAISS
from langchain_huggingface import HuggingFaceEmbeddings
embedding_function =HuggingFaceEmbeddings(
    model_name="sentence-transformers/all-MiniLM-L6-v2"
)
faiss = FAISS.load_local(
            "faiss_db",
            embedding_function,
            "index",
            allow_dangerous_deserialization=True
)
```

https://gist.github.com/ryanchung403/5dad05723846ee93c868355acf1bbcaa

Lab. 自製問答機器人

- 將建立好的向量資料庫匯出
- 建立檔案 qna-bot.py
  - 匯入所需模組
  - 匯入向量資料庫
  - 執行問題相似性搜尋
    - 取得最相似資料
      - 若分數大於1：輸出原問題與「Sorry, I don't know.」
      - 分數小於1：輸出原問題與查詢到的答案

```
Q: What is ship of the desert?
A: The camel is known as the "Ship of the Desert"
```

```
Q: What is the earth?
A: Sorry, I don't know the answer to that question.
```

使用Azure OpenAI Embedding

- 將上例中的Embedding Function改為
  - Azure OpenAI Embedding

requirements.txt

```
langchain_community
pandas
langchain_huggingface
faiss-cpu
sentence-transformers
ipykernel
langchain-openai
```

新增 config.ini

```
[AzureOpenAI]
ENDPOINT = https://xxxxx.xxxxx.azure.com/openai/v1/
KEY = xxxxxx
Embedding_DEPLOYMENT_NAME = xxxxxx
```

新增 app-v2.py

```python
import os
os.environ["KMP_DUPLICATE_LIB_OK"] = "True"
from configparser import ConfigParser
# Set up the config parser
config = ConfigParser()
config.read("config.ini")

from langchain_community.vectorstores import FAISS
import pandas as pd

# Load dataset
animal_data = pd.read_csv("animal-fun-facts-dataset.csv")

# Embedding function - SentenceTransformer - all-MiniLM-L6-v2
# from langchain_huggingface import HuggingFaceEmbeddings
# embedding_function = HuggingFaceEmbeddings(
#     model_name="sentence-transformers/all-MiniLM-L6-v2"
# )

# Embedding function - AzureOpenAI - text-embedding-ada-002
from langchain_openai import OpenAIEmbeddings
embedding_function = OpenAIEmbeddings(
    model=config["AzureOpenAI"]["Embedding_DEPLOYMENT_NAME"],
    base_url=config["AzureOpenAI"]["ENDPOINT"],
    api_key=config["AzureOpenAI"]["KEY"],
)
```

新增 app-v2.py

```
metadatas = []
for i, row in animal_data.iterrows():
    metadatas.append(
        {
            "Animal Name": row["animal_name"],
            "Source URL": row["source"],
            # "Media URL": row["media_link"],
            # "Wikipedia URL": row["wikipedia_link"],
        }
    )

animal_data["text"] = animal_data["text"].astype(str)
faiss = FAISS.from_texts(animal_data["text"].to_list(), embedding_function, metadatas)
# export the model
faiss.save_local("faiss_db_openai", "index")
# import the vector-db from disk
faiss_openai = FAISS.load_local(
    "faiss_db_openai",
    embedding_function,
    "index",
    allow_dangerous_deserialization=True
)

faiss_openai.similarity_search_with_score("What is ship of the desert?", 3)
faiss_openai.similarity_search_with_score("What is the earth?", 3)
```

https://gist.github.com/ryanchung403/c2f20ec6f6d7418b96cc236b65bee95c

檢視結果

```
faiss_openai.similarity_search_with_score("What is ship of the desert?"
```

✓ 0.9s

```
[(Document(id='78186b9e-9948-4a36-9707-a1709d21e52a', metadata={'Animal Name': 'camel', 'S
  np.float32(0.7251117)),
 (Document(id='9e952b82-fa8b-4951-ba9b-30013c1a386c', metadata={'Animal Name': 'gerbil', '
  np.float32(1.1690882)),
 (Document(id='8eef6455-4e7d-4b64-9a0e-afe14f6835aa', metadata={'Animal Name': 'fennec fox
  np.float32(1.2445788))]
```

```
faiss_openai.similarity_search_with_score("What is the earth?", 3)
```

✓ 1.1s

```
[(Document(id='568e8001-a11e-4807-ba61-87201ed65f61', metadata={'Animal Name': 'mouse', 'S
  np.float32(1.2554712)),
 (Document(id='75faf42b-19ae-47d2-aac3-c1c98e8aad5e', metadata={'Animal Name': 'aardvark',
  np.float32(1.2902596)),
 (Document(id='72cde5bc-4e74-4122-9a66-cf41740741fc', metadata={'Animal Name': 'earthworm'
  np.float32(1.3058392))]
```

使用Google Gemini Embedding

- 將上例中的Embedding Function改為
  - Google Gemini Embedding

config.ini

```
[Gemini]
API_KEY = XXXXXXX
```

requirements.txt

```
langchain_community
pandas
langchain_huggingface
faiss-cpu
sentence-transformers
ipykernel
langchain-openai
langchain_google_genai
```

新增 app-gemini.py

```python
import os
os.environ["KMP_DUPLICATE_LIB_OK"] = "True"
from configparser import ConfigParser
# Set up the config parser
config = ConfigParser()
config.read("config.ini")
from langchain_community.vectorstores import FAISS
import pandas as pd
# Load dataset
animal_data = pd.read_csv("animal-fun-facts-dataset.csv")
from langchain_google_genai import GoogleGenerativeAIEmbeddings, ChatGoogleGenerativeAI
embedding_function = GoogleGenerativeAIEmbeddings(
    model="gemini-embedding-001",
    google_api_key=config["Gemini"]["API_KEY"],
)
metadatas = []
for i, row in animal_data.iterrows():
    metadatas.append(
        {
            "Animal Name": row["animal_name"],
            "Source URL": row["source"],
            # "Media URL": row["media_link"],
            # "Wikipedia URL": row["wikipedia_link"],
        }
    )
```

新增 app-gemini.py

```
animal_data["text"] = animal_data["text"].astype(str)

faiss = FAISS.from_texts(animal_data["text"].to_list(), embedding_function, metadatas)

faiss.save_local("faiss_db_gemini", "index")

# import the vector-db from disk

faiss_gemini = FAISS.load_local(
    "faiss_db_gemini", embedding_function, "index", allow_dangerous_deserialization=True
)

faiss_gemini.similarity_search_with_score("What is ship of the desert?", 3)
faiss_gemini.similarity_search_with_score("What is the earth?", 3)
```

https://gist.github.com/ryanchung403/ab879b8cf9e6e11de8683e51c0529a03

檢視結果

```
faiss_gemini.similarity_search_with_score("What is ship of the desert?"
```

✓ 0.5s

```
[(Document(id='3da3db28-18c9-4e18-897c-dae0593c49fc', metadata={'Animal Name': 'camel', 'S
  np.float32(0.33969003)),
 (Document(id='32c1c9da-8150-4708-a9ed-ff40fba37c00', metadata={'Animal Name': 'bactrian c
  np.float32(0.5701376)),
 (Document(id='9a1f2bc6-2ee0-4e45-acf3-7e90d9502a47', metadata={'Animal Name': 'gerbil', '
  np.float32(0.59844303))]
```

```
faiss_gemini.similarity_search_with_score("What is the earth?", 3)
```

✓ 0.3s

```
[(Document(id='24d4591a-7821-41ef-afdc-c053880c80ab', metadata={'Animal Name': 'mouse', 'S
  np.float32(0.6735364)),
 (Document(id='e1aa6df7-af2e-444b-a849-b76b2e01a03e', metadata={'Animal Name': 'honey badg
  np.float32(0.6906671)),
 (Document(id='28a60c45-c421-4f12-98a0-099bfe5725e4', metadata={'Animal Name': 'barn owl',
  np.float32(0.699272))]
```

Embedding Model 比較

	text-embedding-3-small	all-MiniLM-L6-v2	gemini-embedding-001
Dimensions	1536	384	3072
Model-Size	Unknown(in cloud)	0.09 GB	Unknown(in cloud)
Language Support	English, Spanish, French, Chinese	Best for English	over 100 languages

Massive Text Embedding Benchmark https://github.com/embeddings-benchmark/mteb

MTEB

Rank (Bor…	Model	Zero-shot	Memory Us…	Number of P…	Embedding D…	Max Tokens
1	KaLM-Embedding-Gemma3-12B-2511	73%	44884	11.8	3840	32768
2	llama-embed-nemotron-8b	99%	28629	7.5	4096	32768
3	Qwen3-Embedding-8B	99%	14433	7.6	4096	32768
4	gemini-embedding-001	99%			3072	2048
5	Qwen3-Embedding-4B	99%	7671	4.0	2560	32768
6	Octen-Embedding-8B	99%	14433	7.6	4096	32768
7	Seed1.6-embedding-1215	89%			2048	32768
8	Qwen3-Embedding-0.6B	99%	1136	0.596	1024	32768
9	gte-Qwen2-7B-instruct	⚠ NA	29040	7.6	3584	32768
10	Linq-Embed-Mistral	99%	13563	7.1	4096	32768

https://huggingface.co/spaces/mteb/leaderboard

如何選擇 Embedding Model?

- 相似度搜尋效率
  - MTEB
- 模型大小
  - 運算成本、載入速度
- Sequence長度
  - 模型可以允許輸入的token數量
- Dimension 大小
  - 儲存成本
- 語言支援
  - 是否支援中文?

https://supabase.com/blog/fewer-dimensions-are-better-pgvector