

學校:長庚大學

系級:人工智慧學系 大三

學號:B1228022

姓名:梁劍豪 主題:RAG系統打造網站的小客服

Colab:

打造向量資料庫:

https://colab.research.google.com/drive/1yvf6sdL_GYQ3orRzUbmB0yOCq7mZhFsA?usp=sharing

打造RAG系統:

https://colab.research.google.com/drive/1OAyy5rlRWxf5bP3Wv0-UZ1BKTaa_374M?usp=sharing

資料:

長庚大學智慧型語言演算法實驗室網站

<https://ila-lab.github.io/>

重點截圖:

打造向量資料庫:

The screenshot shows a Google Colab notebook titled "RAG01_打造向量資料庫.ipynb". The code cell contains the following Python script:

```
1 #!/usr/bin/env python3
2 import os
3 uploaded_dir = "uploaded_docs"
4 os.makedirs(uploaded_dir, exist_ok=True)
5 print("請將你的 .txt, .pdf, .docx, .html, .js, .py 檔案放到這個資料夾中: " + (uploaded_dir))
```

Below the code cell, there is a note: "請到上傳檔案再繼續。"

The next section, "2. 更新必要套件並引入", shows the output of running pip install commands:

```
1 !pip install -U langchain langchain-community pyPDF python-docx faiss-cpu
2 !pip install -U sentence-transformers transformers
```

The output shows the installation of various packages, including langchain, langchain-community, pyPDF, python-docx, faiss-cpu, sentence-transformers, and transformers.

```
1 from langchain_community.document_loaders import TextLoader, PyPDFLoader, UnstructuredWordDocumentLoader
2 from langchain_text_splitters import RecursiveCharacterTextSplitter
3 from langchain_community.embeddings import HuggingFaceEmbeddings
4 from langchain_community.vectorstores import FAISS
```

3. 依 Google 建議加入 EmbeddingGemma 前綴

Google 建議，在文本部份的 Embedding 要用以下格式：

```
title: {title|none} | text: ...
```

而問題 Query 要用：

```
Query: task: search result | query: ...
```

```
1 from langchain_community.embeddings import HuggingFaceEmbeddings
```

```
1
2 class EmbeddingGemmaEmbeddings(HuggingFaceEmbeddings):
3     def __init__(self, **kwargs):
4         super().__init__(
5             model_name="google/embeddinggemma-200m",      # HF 上的官方模型
6             encode_kwargs={"normalize_embeddings": True},    # 一般檢索慣例
7             **kwargs
8         )
9
10    def embed_documents(self, texts):
11        # 文件問題：title 可用 "none"，或自行帶入檔名/章節標題以做標加分
12        texts = [f"title: none | text: {t}" for t in texts]
13        return super().embed_documents(texts)
14
15    def embed_query(self, text):
16        # 查詢前綴：官方建議的 Retrieval-Query 前綴
17        return super().embed_query(f"task: search result | query: {text}")
```

```
1 pip install unstructured
```

```
Collecting unstructured
  Downloading unstructured-0.18.15-py3-none-any.whl.metadata (24 kB)
Requirement already satisfied: charset-normalizer in /usr/local/lib/python3.12/dist-packages (from unstructured) (3.4.4)
Collecting filetype (from unstructured)
  Downloading filetype-1.2.0-py2.py3-none-any.whl.metadata (6.5 kB)
Collecting python-magic (from unstructured)
```

```
1
2
3 # 引入適合 HTML, JS, Python 檔案的載入器
4 # 使用 UnstructuredFileLoader 處理多種文件（需要安裝相關倚賴）
5 from langchain_community.document_loaders import TextLoader, PyPDFLoader, UnstructuredWordDocumentLoader, UnstructuredFileLoader
6
7 upload_dir = "uploaded_docs"
8 os.makedirs(upload_dir, exist_ok=True)
9 # 更新提示訊息
10 print("請將你的 .txt, .pdf, .docx, .html, .js, .py 檔案放到這個資料夾中: " + upload_dir)
11
12 # ... (EmbeddingGemmaEmbeddings 類別保持不變)
13
14 folder_path = upload_dir
15 documents = []
16 for file in os.listdir(folder_path):
17     path = os.path.join(folder_path, file)
18
19     # 略過隱藏檔案 (如 .DS_Store)
20     if file.startswith('.'):
21         continue
22
23     # 對不同檔案型態使用適當的載入器
24     if file.endswith('.txt'):
25         loader = TextLoader(path)
26     elif file.endswith('.pdf'):
27         loader = PyPDFLoader(path)
28     elif file.endswith('.docx'):
29         loader = UnstructuredWordDocumentLoader(path)
30     # 新增對 .html, .js, .py 檔案的處理
31     elif file.endswith(('.html', '.js', '.py')):
32         # UnstructuredFileLoader 可以處理多種純文字和結構化文件
33         # 它會自動嘗試提取內容，特別適合程式碼和結構化檔案
34         loader = UnstructuredFileLoader(path)
35     else:
36         continue
37
38     try:
39         documents.extend(loader.load())
40     except Exception as e:
41         print(f"Error loading file {file}: {e}")
```

hw7 - Google 請編輯器 × RAGO1_打造向量資料庫.ipynb × cpu_b1228022.ai3_集訓課_... × +

← → ⌂ colab.research.google.com/drive/1yvf6sdL_GYQ3orRzUbmB0yOCq7mZhfA

RAGO1_打造向量資料庫.ipynb ☆ ↗

檔案 編輯 檢視畫面 插入 執行階段 工具 說明

指令 + 程式碼 + 文字 ▶ 全部執行

```

1 38     try:
2 39         documents.extend(loader.load())
3 40     except Exception as e:
4 41         print(f"處理檔案 {file} 時發生錯誤: {e}")
4 42         continue
4 43

```

請將你的 .txt, .pdf, .docx, .html, .js, .py 檔案放到這個資料夾中: uploaded_docs

□ 5. 建立向量資料庫

```

1 splitter = RecursiveCharacterTextSplitter(chunk_size=500, chunk_overlap=100)
2 split_docs = splitter.split_documents(documents)

```

```

1 from google.colab import userdata
2 hf_token = userdata.get("huggingface")

```

```

1 from huggingface_hub import login
2 login(token=hf_token)

```

```

1 embedding_model = HuggingFaceEmbeddings(model_name="all-MiniLM-L6-v2")
2 vectorstore = FAISS.from_documents(split_docs, embedding_model)

```

/usr/local/lib/python3.12/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret 'HF_TOKEN' does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (<https://huggingface.co/settings/tokens>), set it as secret in your Google Colab and restart your session.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.

warnings.warn(
modules.json: 100% [0:00<0:00, 11.4kB/s]
config_sentence_transformers.json: 100% [0:00<0:00, 4.57kB/s]
README.md: 10.5k? [0:00<0:00, 299kB/s]
sentence_bert_config.json: 100% [0:00<0:00, 4.58kB/s]

21:15 27/10/2025

□ 編號 終端機

21°C 多雲

搜尋 檻

21:16 27/10/2025

hw7 - Google 請編輯器 × RAGO1_打造向量資料庫.ipynb × colab.research.google.com/drive/1yvf6sdL_GYQ3orRzUbmB0yOCq7mZhfA

RAGO1_打造向量資料庫.ipynb ☆ ↗

檔案 編輯 檢視畫面 插入 執行階段 工具 說明

指令 + 程式碼 + 文字 ▶ 全部執行

To authenticate with the Hugging Face Hub, create a token in your settings tab (<https://huggingface.co/settings/tokens>), set it as secret in your Google Colab and restart your session.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.

warnings.warn(
modules.json: 100% [0:00<0:00, 11.4kB/s]
config_sentence_transformers.json: 100% [0:00<0:00, 4.57kB/s]
README.md: 10.5k? [0:00<0:00, 299kB/s]
sentence_bert_config.json: 100% [0:00<0:00, 4.58kB/s]
config.json: 100% [0:00<0:00, 24.2kB/s]
model_safetensors: 100% [0:00<0:00, 65.2MB/s]
tokenizer_config.json: 100% [0:00<0:00, 16.8kB/s]
vocab.txt: 232k? [0:00<0:00, 8.08MB/s]
tokenizer.json: 466k? [0:00<0:00, 28.7MB/s]
special_tokens_map.json: 100% [0:00<0:00, 5.12kB/s]
config.json: 100% [0:00<0:00, 21.8kB/s]

21:16 27/10/2025

□ 編號 終端機

21°C 多雲

搜尋 檻

21:16 27/10/2025

▼ 6. 儲存向量資料庫

```

1 vectorstore.save_local("faiss_db")

```

```

1 zip -r faiss_db.zip faiss_db
adding: faiss_db/ (stored 0%)
adding: faiss_db/index.faiss (deflated 7%)
adding: faiss_db/index.pkl (deflated 61%)

```

```

1 print("壓縮好的向量資料庫已儲存為 'faiss_db.zip'，請下載此檔案備份。")
2 壓縮好的向量資料庫已儲存為 'faiss_db.zip'，請下載此檔案備份。

```

+ 程式碼 + 文字

21:16 27/10/2025

打造RAG系統:

hw7 - Google 論壇硬體

RAG02_打造_RAG_系統.ipynb

cpu_b1228022.ai3.要到哪.jso

0. 讀入 vector dataset

```
1 !pip -q install gdown
```

Traceback (most recent call last):
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/cli/base_command.py", line 179, in exc_logging_wrapper
 status = run_func(*args)
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/cli/req_command.py", line 67, in wrapper
 return func(self, options, args)
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/commands/install.py", line 447, in run
 conflicts = self._determine_conflicts(to_install)
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/commands/install.py", line 578, in _determine_conflicts
 return check_install_conflicts(to_install)
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/operations/check.py", line 101, in check_install_conflicts
 package_set = _create_package_set_from_installed()
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/operations/check.py", line 39, in create_package_set_from_installed
 for dist in envir.inited_distributions(local_only=False, skip=()):
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/metadata/base.py", line 664, in <genexpr>
 return (d for d in it if d.canonical_name not in skip)
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/metadata/base.py", line 612, in iter_all_distributions
 for dist in self._iter_distributions()
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/metadata/importlib_envs.py", line 176, in _iter_distributions
 yield from finder.find_location()
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/metadata/importlib_envs.py", line 79, in find
 for dist, info_location in self._find_implementations():
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/metadata/importlib_envs.py", line 64, in _find_implementations
 raw_name = get_dist_name(dist)
File "/usr/local/lib/python3.12/dist-packages/pip/_internal/metadata/importlib_compat.py", line 52, in get_dist_name
 name = cast(Any, dist).name
File "/usr/lib/python3.12/importlib/metadata/_init_.py", line 457, in name
 return self.metadata['Name']

終端機

21°C 多雲

搜尋 檻 瀏覽 地址 帳單 檔案 編輯 檢視畫面 插入 執行階段 工具 說明

指令 程式碼 文字 全部執行

分享

逐條

hw7 - Google 論壇硬體

RAG02_打造_RAG_系統.ipynb

cpu_b1228022.ai3.要到哪.jso

1 GDRIVE_PUBLIC_URL = "https://drive.google.com/file/d/1ElfIuypYmD8BPh-PbR0gUjkzDws_16/view?usp=sharing"

```
1 !gdown --fuzzy -O faiss_db.zip "(GDRIVE_PUBLIC_URL)"
```

Downloading
From: https://drive.google.com/u/1/?id=1ElfIuypYmD8BPh-PbR0gUjkzDws_16
To: /content/faiss_db.zip
100% 44.4k/44.4k [00:00:00.00, 40.6MB/s]

```
1 !unzip faiss_db.zip
```

Archive: faiss_db.zip
replace faiss_db/index/faiss? [y]es, [n]o, [A]ll, [N]one, [r]ename: A
inflating: faiss_db/index/faiss
inflating: faiss_db/index.pkl

1. 安裝並引入必要套件

```
1 !pip install -U langchain langchain-community faiss-cpu transformers sentence-transformers huggingface_hub
```

```
Requirement already satisfied: langchain in /usr/local/lib/python3.12/dist-packages (0.3.27)
Collecting langchain
  Downloading langchain-1.0.2-py3-none-any.whl.metadata (4.7 kB)
Collecting langchain-community
  Downloading langchain_community-0.4-py3-none-any.whl.metadata (3.0 kB)
Collecting faiss-cpu
  Downloading faiss-cpu-1.12.0-cp312-cp312-manylinux_2_27_x86_64.manylinux_2_28_x86_64.whl.metadata (5.1 kB)
Requirement already satisfied: transformers in /usr/local/lib/python3.12/dist-packages (4.57.1)
Requirement already satisfied: sentence-transformers in /usr/local/lib/python3.12/dist-packages (5.1.1)
Collecting sentence-transformers
  Downloading sentence_transformers-5.1.2-py3-none-any.whl.metadata (16 kB)
Requirements already satisfied: huggingface_hub in /usr/local/lib/python3.12/dist-packages (0.35.3)
Collecting huggingface_hub
  Downloading huggingface_hub-0.35.3-py3-none-any.whl.metadata (14 kB)
Collecting langchain
  Downloading langchain-2.0.0-py3-none-any.whl.metadata (3.5 kB)
Collecting langchain_community-0.4.0-py3-none-any.whl.metadata (7.4 kB)
Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in /usr/local/lib/python3.12/dist-packages (from langchain) (2.11.10)
```

終端機

21°C 多雲

搜尋 檻 瀏覽 地址 帳單 檔案 編輯 檢視畫面 插入 執行階段 工具 說明

指令 程式碼 文字 全部執行

分享

逐條

```

1 from langchain_community.vectorstores import FAISS
2 from langchain_community.embeddings import HuggingFaceEmbeddings
3 from langchain_community.chat_models import ChatOpenAI

1 from openai import OpenAI
2 import gradio as gr

2. 自訂 E5 embedding 類別

1 import os
2 from google.colab import userdata

1 hf_token = userdata.get('HuggingFace')

1 from huggingface_hub import login
2 login(token=hf_token)

1 class EmbeddingGemaEmbeddings(HuggingFaceEmbeddings):
2     def __init__(self, **kwargs):
3         super().__init__(
4             model_name='google/embeddinggema-300m',
5             encode_kwargs={'normalize_embeddings': True},
6             **kwargs
7         )
8
9     def embed_documents(self, texts):
10        # 你也可以把 "none" 改成真實標題《標名/章節名》，效果會更棒
11        texts = [{"title": None, "text": [t]} for t in texts]
12        return super().embed_documents(texts)
13
14    def embed_query(self, text):
15        # 官方檢索建議前綴
16        return super().embed_query(f'task: search_result | query: {text}')

```

3. 載入 faiss_db

```

1 embedding_model = HuggingFaceEmbeddings(model_name="sentence-transformers/all-MiniLM-L6-v2")
2 vectorstore = FAISS.load_local(
3     "faiss_db",
4     embeddings=embedding_model,
5     allow_dangerous_deserialization=True
6 )

1 retriever = vectorstore.as_retriever(search_kwargs={"k": 4})

```

4. 設定好要的 LLM

```

1 import aiosuite as ai
1 api_key = userdata.get('Groq')
1 os.environ['GROQ_API_KEY']=api_key

Groq 中 llama-3.3-70b-versatile模型 好像比較適合理解程式碼和 RAG 任務的。
1 model = "groq:llama-3.3-70b-versatile"
1 client = ai.Client()

```

5. prompt 設計

```

1 system_prompt = """
2 你現在是「長庚大學人工智能學系 智慧型語言演算法實驗室 (CGU-AI-SILA Lab)」的 AI 學生小客服。
3
4 你的說話風格必須：
5 1. **超級貼切、熱情，且口吻自然**：就像一位熱心的學長姐。

```

人設/背景設定：

hw7 - Google 異端硬體 x RAG02_打造_RAG_系統.ipynb x cpu_b1228022.ai3_桌面上... x +

colab.research.google.com/drive/1OAyy5rlRWxf5bP3Wv0-UZ1BKtaa_374M

RAG02_打造_RAG_系統.ipynb ☆

檔案 編輯 檢視畫面 插入 執行劇段 工具 說明

指令 + 程式碼 + 文字 ▶ 全部執行

5. prompt 設計

```
1 system_prompt = """
2 你現在是「長庚大學人工智慧學系 智慧型語言演算法實驗室 (CGU-AI-SILA Lab)」的 AI 學生小客服。
3
4 你的說話風格必須：
5 1. **超級親切、熱情，且口吻自然**；就像一位熱心的學長姐。
6 2. **使用台灣習慣的中文**；繁體中文、語氣活潑、自然。
7 3. **簡潔且附帶具體建議**；回答要點明確，並给出真實的下一步建議。
8 4. **嚴格依據提供的「資料」來回答學生的問題**。
9
10 你的回答規範：
11 * **回答結構**：必須包含「親切的稱呼（如：同學、您好）」、「簡潔的回答」和「具體的建議或下一步指引」。
12 * **連結處理**：如果提供資料（`retrieved_chunks`）中有網址，請務必使用 `#Markdown` 格式的超連結「[連結文字](網址)** 来呈現，讓同學可以直接點擊。
13 * **資料不足時**：請主動建議同學可以「寄信詢問實驗室辦公室」或「直接請教人工智慧學系的老師」，強調具體聯絡方式。
14 """
15
16 prompt_template = """
17 親愛的同學，您好！我是長庚大學人工智慧學系的小客服，很高興為您服務！
18
19 請根據下列實驗室資料來回答問題：
20 [retrieved_chunks]
21
22 學生的問題是：{question}
23
24 ---
25
26 請注意：
27 1. **只使用上面 [retrieved_chunks] 內的資訊來回答**。
28 2. **重要**：如果資料中有提供網址或連結，請務必將其轉換為 #Markdown 格式的超連結「[連結名稱](網址)**，例如：[長庚大學首頁](https://www.cgu.edu.tw/)。
29 3. 若資料不足，請建議他們可以**直接寄信詢問實驗室辦公室**或**直接請教人工智慧學系的老師**。
30
31 ---
32
33 請用**台灣習慣的中文**，依照上述規範開始回覆。
34 """

```

+ 程式碼 + 文字

終端機

21:18 27/10/2025

hw7 - Google 異端硬體 x RAG02_打造_RAG_系統.ipynb x cpu_b1228022.ai3_桌面上... x +

colab.research.google.com/drive/1OAyy5rlRWxf5bP3Wv0-UZ1BKtaa_374M#scrollTo=pWFDUb3mD-6X

RAG02_打造_RAG_系統.ipynb ☆

檔案 編輯 檢視畫面 插入 執行劇段 工具 說明

指令 + 程式碼 + 文字 ▶ 全部執行

6. 使用 RAG 來回應

搜尋與使用者問題相關的資訊，根據 prompt 樣版去讓 LLM 回應。

```
1 chat_history = []
2
3 def chat_with_rag(user_input):
4     global chat_history
5     # 取回先前資訊
6     docs = retriever.invoke(user_input)
7
8     retrieved_chunks = "\n\n".join(doc.page_content for doc in docs)
9
10    # 將自定 prompt 套入格式
11    final_prompt = prompt_template.format(retrieved_chunks=retrieved_chunks, question=user_input)
12
13    # 用 AI Suite 呼叫語言模型
14    response = client.chat.completions.create(
15        model=model,
16        messages=[
17            {"role": "system", "content": system_prompt},
18            {"role": "user", "content": final_prompt},
19        ]
20    )
21    answer = response.choices[0].message.content
22
23    chat_history.append((user_input, answer))
24
25    return answer
26
```

+ 程式碼 + 文字

7. 用 Gradio 打造 Web App

```
1 with gr.Blocks() as demo:
2     gr.Markdown("#智慧型語言演算法實驗室小客服")
3     chatbot = gr.Chatbot()
4     msg = gr.Textbox(placeholder="請輸入您的問題... ")
5
6     def respond(message, chat_history_local):

```

終端機

21:19 27/10/2025

```
1 with gr.Blocks() as demo:
2     gr.Markdown("#智慧型語言演算法實驗室小客服")
3     chatbot = gr.Chatbot()
4     msg = gr.Textbox(placeholder="請輸入你的問題...")
```

/tmp/ipython-input-2476923104.py:3: UserWarning: You have not specified a value for the 'type' parameter. Defaulting to the 'tuples' format for chatbot messages, but this is deprecated and will be removed in future versions.

chatbot = gr.Chatbot()

It looks like you are running Gradio on a hosted Jupyter notebook, which requires 'share=True'. Automatically setting 'share=True' (you can turn this off by setting 'share=False' in 'launch()').

Colab notebook detected. This cell will run indefinitely so that you can see errors and logs. To turn off, set debug=False in launch().

* Running on public URL: <https://1ed4179fd2e2c2a664c.gradio.live>

This share link expires in 1 week. For free permanent hosting and GPU upgrades, run 'gradio deploy' from the terminal in the working directory to deploy to Hugging Face Spaces (<https://huggingface.co/spaces>)

 No interface is running right now

問題

請問『智慧型語言演算法實驗室』主要的研究方向是什麼？有沒有相關網頁可以看？

研究團隊有誰？總共有多少人？

教授有個人網頁嗎？可以直接給連結給我嗎？

教授有GitHub嗎？可以直接給教授GitHub連結我嗎？

論文共有多少？顯示全部

有沒有"Relation-Aware Image Captioning with Hybrid-Attention for Explainable Visual Question Answering"相關網頁？

論文可以哪裡找？

聯絡地址可以在哪裡找？

```
1 ref: https://docs.langchain.com/oss/python/migrate/langchain-v1#simplified-package
2
```

Gradio 的對話結果：

向量資料庫(0):

有js, html

https://drive.google.com/file/d/1dLI-xvdfvl5sR0PSI8MdM420clzsilrC/view?usp=drive_link

忘記了截圖，但輸出大概問什麼都總是只有一半，例如研究團隊只顯示到六個左右、問連結無法讀取、論文只有2個等等。

猜測是只讀取到部分資料，如果連同js的話，會太大，而且有很多不相幹的東西。

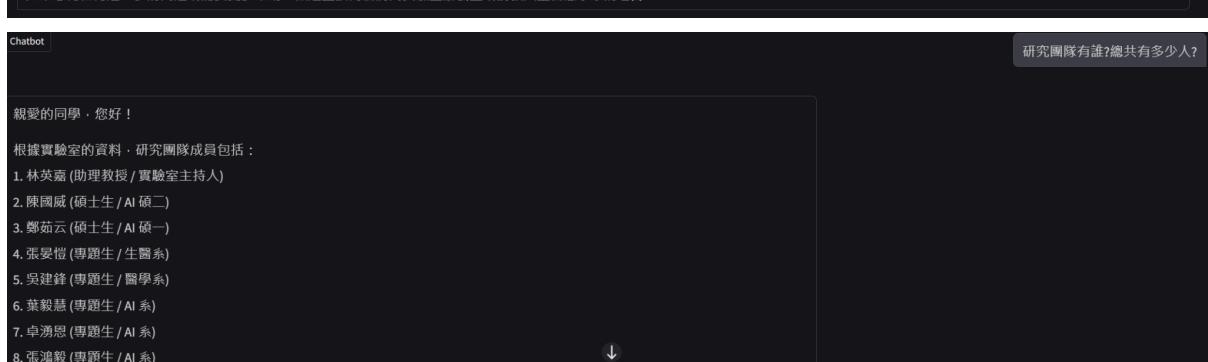
向量資料庫(1):

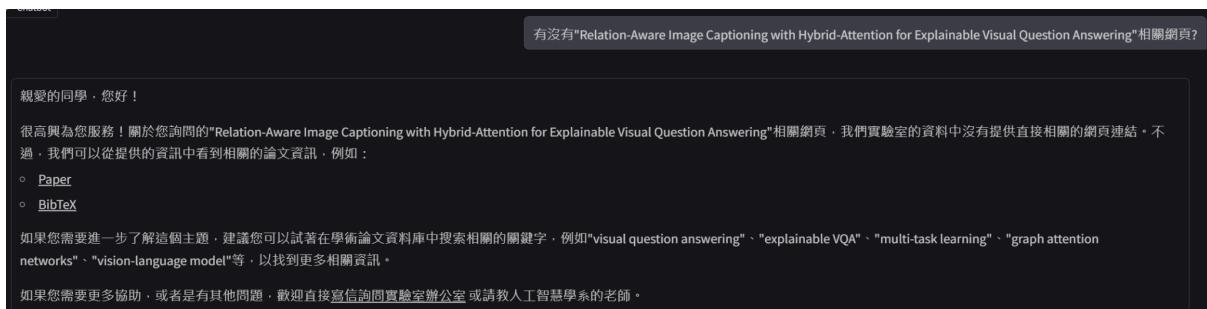
只有html

https://drive.google.com/file/d/1YkKWT_g-viCdySdTYswT9xQhJeD-QXf9/view?usp=drive_link

大部分問題都對，比上一個好，有連結，但有部分是錯的連結，而且論文的部分還是未修改到。

猜測還是只讀取到部分資料，盡量精簡。





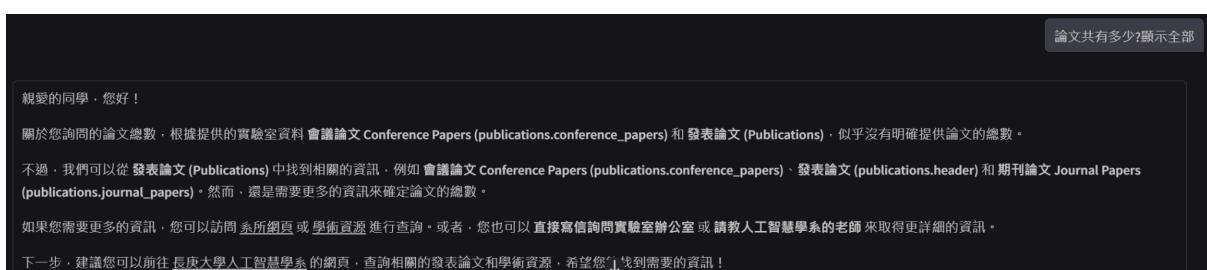
向量資料庫(2):

從html取出文字以及大部分連結變成txt

https://drive.google.com/file/d/1HdbrzbCN23_tzdPzQkYJc2m7FgAkLW2b/view?usp=drive_link

雖然有連結，但不是全部都對，而且問的問題回答不了。

猜測因為有部分不必要的東西，例如可能成員後面還會接一個class的表示那樣。



向量資料庫(3)(最終):

再整理過txt

https://drive.google.com/file/d/1E1fluvpYmDHBHFn-PbRUgUjkzD8ws_l6/view?usp=drive_link

雖然有少部分回答不到問題，但大部分問題都對的還有對的連結，而且還有步驟地教你怎樣查閱。

請問『智慧型語言演算法實驗室』主要的研究方向是什麼？有沒有相關網頁可以看？

親愛的同學，您好！

我們的實驗室——智慧型語言演算法實驗室，主要的研究方向是自然語言處理、大型語言模型和 Medical NLP。如果您想要了解更多關於我們的研究方向，歡迎點擊研究主題頁面。

我們的實驗室致力於結合深度類神經網路和大型語言模型的研究，並與林口長庚醫院和國立陽明交通大學密切合作，以跨學科的視角解決複雜問題。

如果您想要更深入地了解我們的研究，歡迎點擊發表論文頁面。

若您有任何問題或需要更多資訊，歡迎聯絡我們，電子郵件是 ilalab.cgu@gmail.com，電話是 03-2118800 #2547。

若您需要詢問實驗室的研究進展，也可以直接寫信詢問實驗室辦公室或直接請教人工智慧學系的老師，我們會盡力協助您。

論文可以哪裡找？

親愛的同學，您好！

很高興為您服務！根據實驗室的資訊，您可以在 發表論文 (Publications) 部分找到論文相關的資訊，包括 期刊論文 Journal Papers 和 會議論文 Conference Papers。如果您想要瀏覽這些論文的詳細內容，建議您點擊 發表論文 連結來查看更多資訊。

另外，如果您需要更多的資源，實驗室也提供了 大學官網 和 系所網頁 的連結，您可以點擊這些連結來瀏覽更多的資訊。

如果您仍然有問題或需要更多的幫助，建議您可以直接寫信詢問實驗室辦公室或請教人工智慧學系的老師。電子郵件地址是 ilalab.cgu@gmail.com，電話號碼是 03-2118800 #2547。我們會盡量協助您解決問題！

教授有GitHub嗎？可以直接給教授GitHub連結我嗎？

親愛的同學，您好！

根據實驗室的資料，似乎沒有直接提供教授的 GitHub 連結。如果您想要找到教授的 GitHub 連結，建議您可以：

1. 聯絡我們：您可以透過電子郵件 ilalab.cgu@gmail.com 或電話 03-2118800 #2547 聯繫實驗室的辦公室，詢問是否有教授的 GitHub 連結。
2. 查找系所網頁：您也可以到 系所網頁 查看是否有教授的個人頁面或研究團隊的介紹，可能會有相關的連結。
3. 直接詢問老師：如果您有老師的聯繫方式，直接詢問老師是否有 GitHub 連結也是個好方法。

希望這些建議對您有幫助，如果您有任何進一步的問題，請不要猶豫地問我喲！