1. AI生成數據分析報告

自動生成客戶委託分析報告、網站提供的定期分析報告。

* 呼叫別人的免費或收費AI服務

呼叫Gemini, OpenAI, Grok, Copilot, DeepSeek..

* 呼叫自己架設的AI服務:

ollama

1. Ollama介紹

Ollama 是一個開源工具，用於在本地運行大型語言模型 (LLM)。它具有以下主要特點：

1. **本地運行** - 允許在個人電腦或伺服器上部署和運行語言模型，無需雲端服務
2. **支援多種模型** - 從您的代碼中可以看到支援如 Gemma、Llama3、DeepSeek 等模型
3. **API 接口** - 提供簡單的 API 來與模型互動，如您的代碼中使用的 [generate()](vscode-file://vscode-app/c:/Users/clhuang/AppData/Local/Programs/Microsoft%20VS%20Code%20Insiders/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html) 和 [chat()](vscode-file://vscode-app/c:/Users/clhuang/AppData/Local/Programs/Microsoft%20VS%20Code%20Insiders/resources/app/out/vs/code/electron-sandbox/workbench/workbench.html) 方法
4. **串流回應** - 支援流式回應模式 (streaming mode)，可逐步返回生成內容
5. 呼叫方式

|  |
| --- |
| url = "http://163.18.22.32:11435/api/generate"  # 設置遠程 Ollama 模型的基礎 URL  REMOTE\_OLLAMA\_URL = "http://163.18.22.32:11435"  model\_name = "gemma3:4b" # 默認模型名稱  # model\_name = "qwen2.5:7b" # 默認模型名稱  #model\_name = "deepseek-r1:14b" # 默認模型名稱  @csrf\_exempt  def api\_get\_userkey\_llm\_report(request):    result = get\_userkey\_data(request)  # Check if result is an error dictionary  if isinstance(result, dict) and 'error' in result:  return JsonResponse(result)    response\_from\_userkeyword, response\_from\_sentiment = result    userkey = request.POST.get('userkey')  key\_occurrence\_cat = response\_from\_userkeyword['key\_occurrence\_cat']  key\_time\_freq = response\_from\_userkeyword['key\_time\_freq']  key\_freq\_cat = response\_from\_userkeyword['key\_freq\_cat']    sentiCount = response\_from\_sentiment['sentiCount']  line\_data\_pos = response\_from\_sentiment['data\_pos']  line\_data\_neg = response\_from\_sentiment['data\_neg']    # print(response1\_data)  # 系統提示指令  system\_prompt = f"以下是有關於[{userkey}]的網路聲量資訊，請做一份至少500字的詳細的專業分析報告。請使用繁體中文，並使用Markdown語法。"  # 都出所有的輸入提示詞  prompt = f'''{system\_prompt}\n\n  (1)聲量分析: 根據以下資料，幫我撰寫一份至少500字的詳細的專業分析報告  以下是熱門程度，有多篇新聞報導提到:\n\n{key\_occurrence\_cat}\n\n  以下是時間趨勢，這個關鍵字在過去幾天的報導數量變化:\n\n{key\_time\_freq}\n\n  (2)情緒分析: 請根據以下資料，幫我撰寫一份至少500字的詳細的專業分析報告  以下是情緒分析比率，正面負面的分布情況:\n\n{sentiCount}\n\n  以下是情緒變化的時間趨勢，在過去幾天的報導情緒正負面的篇數數量變化:\n\n{line\_data\_pos}\n\n{line\_data\_neg}\n\n  (3)分析的內容包括但不限於以下幾個方面：  標題  摘要  關鍵字  內容  建議  總結:  '''  print(prompt)      # 這裡你可以呼叫ChatGPT的API來生成報告，或其他任何AI大型模型的API  # 這裡使用requests來呼叫我用Ollama架設的遠端的API  payload = {  "model": model\_name,  "prompt": prompt,  "stream": False  }  try:  response = requests.post(url, json=payload, timeout=100) # Add a timeout  result = response.json()  print(result['response'])  except:  print("Error:", response.status\_code, response.text)  return JsonResponse({'error': 'Failed to generate report. Please try again later.'})    # 取得AI生成的報告  response\_report = {  'report': result['response']  #'report': markdown.markdown(result['response'])  }    # Combine dictionaries correctly  return JsonResponse(response\_report) |