

# AI 旅遊通

## AI Travel Guide

學生：陳楷茗、陳士軒、黃豐棋

指導老師：辛錫進 教授

國立聯合大學 資訊工程學系

苗栗市南勢里聯大2號

[u1124017@o365.nuu.edu.tw](mailto:u1124017@o365.nuu.edu.tw)

### 摘要

為解決自助旅遊資訊破碎化及生成式 AI 常見之幻覺問題，本專題開發並整合大型語言模型 (LLM) 與檢索增強生成 (RAG) 技術的「AI 旅遊通」系統，利用 ChromaDB 驗證景點真實性，並串接 TDX 與 CWA API 以獲取即時交通與天氣資訊。前端結合 Gradio 網頁與具備 GPS 定位之 Android App，實現周邊景點搜尋與地圖整合。結果顯示，本系統能有效整合多元數據，提供準確、即時且個人化的行程規劃建議。

關鍵詞：大型語言模型、檢索增強生成、ChromaDB、TDX API、Android 開發。

### Abstract

To address fragmented travel information and AI hallucinations, this project presents "AI Travel Guide," a system integrating LLM and RAG technology. Using ChromaDB for factual verification and TDX/CWA APIs for real-time transport and weather data, the system ensures itinerary accuracy. Featuring Gradio and Android interfaces with GPS integration, it provides personalized, real-time, and context-aware travel planning.

Keywords: Large Language Model (LLM), Retrieval-Augmented Generation (RAG), ChromaDB, TDX API, Android Development.

### 一、緒論

當旅客嘗試規劃一趟旅程時，往往發現旅遊資訊散落在不同的平台與應用程式中。旅客可能需要在 Instagram 或部落格尋找熱門景點，切換至 Google Maps 確認地理位置，再開啟台鐵或客運 App 查詢交通班次，最後還需至中央氣象署網站確認天氣狀況。這種在多個視窗與 App 之間來回切換的過程不僅繁瑣，更常因資訊銜接不順暢（例如：查到了景點卻發現無合適大眾運輸可達）而導致行程安排的挫折。

現有的通用型 AI 模型在旅遊領域存在兩個主要缺陷：首先是「幻覺」（Hallucination）問題，模型可能一本正經地編造出根本不存在的景點或錯誤的交通路線，這對依賴大眾運輸的旅客而言是極大的風險；其次是缺乏「即時性」，模型無法得知下一班公車的確切時間，也無法掌握當地的即時天氣變化。

針對上述問題，本專題旨在開發一個基於大型語言模型 (LLM) 的旅遊建議助手原型系統。主要目的包括：

1. 個人化建議生成：透過分析使用者偏好（如預算、喜好、同行對象），提供高度客製化的行程規劃。
2. 異質資料整合：有效整合結構化

的開放資料（如景點、交通、天氣）與非結構化的使用者評論，作為 LLM 的知識基礎。

3. 即時資訊增強：讓 AI 助手能夠存取最新的天氣預報與交通動態，確保建議的時效性。
4. 事實查核機制：利用向量資料庫技術檢索真實存在的景點資料，降低 AI 幻覺風險。
5. 跨平台前端實作：建構直覺的網頁介面與功能完整的 Android 行動應用程式，提供無縫的使用者體驗。

## 二、系統架構與設計

本系統採用前後端分離的架構設計，核心邏輯以 Python 3.10 開發，利用其豐富的 AI 生態系與資料處理套件。

### 2.1 系統開發環境

在技術選型上，本專題採用以下關鍵工具：

1. Gradio：作為網頁端介面框架，用於快速建構機器學習模型的互動演示環境。
2. Hugging Face Router API：考量地端硬體資源限制，透過雲端 API 呼叫 Qwen2.5-7B-Instruct 模型 [9]。該模型具備優秀的中文理解與生成能力，適合處理旅遊相關的複雜查詢。
3. ChromaDB：選用此輕量級向量資料庫儲存景點嵌入向量，支援高效的語意搜尋。
4. Android Studio：用於開發行動端應用程式，整合 Java 與 Android SDK 實作原生功能。

### 2.2 資料前處理與向量資料庫

為確保 AI 能夠依據事實回答，本專題使用了交通部觀光署的開放資料集 [3]。在資料匯入前，進行了嚴格的 ETL (Extract, Transform, Load) 程序：

1. 資料清洗：剔除欄位缺失或格式錯誤的項目。
2. 文本合併：將分散的欄位（如景點名稱、介紹、地址、開放時間）合併為一段完整的描述文本，以利模型理解上下文。
3. 向量嵌入 (Embedding)：利用 sentence-transformers 套件中的 paraphrase-multilingual-MiniLM-L12-v2 模型，將文本轉換為 384 維度的數值向量，並存入 ChromaDB 建立索引。此步驟是實現「語意搜尋」的關鍵，讓系統能理解「古蹟」與「歷史建築」的語意關聯。

### 2.3 資料流與 API 整合

本系統強調數據的即時性，整合了多個外部 API：

1. TDX 運輸資料流通服務：作為核心交通引擎，取代需付費的 Google Maps API。系統使用 /v2/Trip/Planning 介面進行跨運具（台鐵、公車、捷運）路徑規劃，並使用 /Rail/TRA/Station 與 /Bus/Stop 介面進行站點定位 [2]。
2. 中央氣象署 (CWA) API：使用 F-C0032-001 一般天氣預報資料 [1]，解析 JSON 回傳值中的 Wx (天氣現象) 與 PoP (降雨機率) 參數，讓 AI 能根據天氣狀況給出穿著或行程調整建議（例如：下雨機率高時推薦室內景點）。

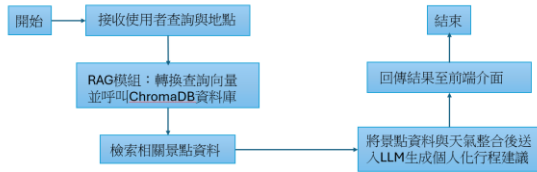


圖1:系統流程圖

### 三、關鍵技術與理論

#### 3.1 檢索增強生成 (RAG) 技術

本專題採用 RAG (Retrieval-Augmented Generation) 架構來解決 LLM 的幻覺問題。其標準化流程如下：

1. 檢索 (Retrieve)：當使用者輸入「想去海邊」時，系統將查詢語句轉換為向量，並在 ChromaDB 中進行近似最近鄰搜索 (Approximate Nearest Neighbor Search)，找出語意相似度最高的前 K 筆景點資料。
2. 增強 (Augment)：系統將檢索到的靜態景點資訊，結合即時呼叫 API 取得的動態天氣與交通數據，注入到預設的 Prompt Template 中。
3. 生成 (Generate)：將包含豐富上下文 (Context) 的 Prompt 輸入至 LLM，引導模型扮演「專業導遊」的角色，依據提供的事實資料生成建議。

#### 3.2 向量嵌入與語意搜尋

為了讓電腦理解人類語言的深層含義，本系統導入自然語言處理 (NLP) 中的 Embedding 技術。選用的 paraphrase-multilingual-MiniLM-L12-v2 模型經過多語言語料庫預訓練，能將不同詞彙映射到同一向量空間。系統計算使用者查詢向量與資料庫景點向量之間的「餘弦相似度」(Cosine Similarity)，數值越接近 1 代表語意越相關。這使得系統能處理模糊查詢（如：「適合

親子同遊的地方」），而非僅限於傳統的關鍵字精確比對。

#### 3.3 智慧交通路徑規劃邏輯

交通路徑規劃是本系統最具挑戰性的部分。由於 TDX API 需要精確的起訖點座標，本專題實作了一套轉換邏輯：

1. 站點映射演算法 (Station Mapping)：當使用者輸入行政區（如「苗栗縣」）時，系統透過模糊比對將其對應至該區域的主要交通節點（如「苗栗火車站」的經緯度），作為路徑規劃的起點。
2. 多模態路徑試算：系統將起訖點座標帶入 TDX API，參數設定涵蓋多種大眾運輸工具。程式會解析回傳的複雜 JSON 結構，提取「總移動時間」、「轉乘次數」與「預估票價」，並過濾掉轉乘次數過多（>3 次）的不合理方案。
3. Token 快取機制：針對 TDX API 的呼叫頻率限制，本專題實作了 Token 管理機制。系統會在記憶體中快取 Access Token，僅在 Token 過期時才重新向驗證伺服器申請，有效降低 API 延遲並避免 HTTP 429 (Too Many Requests) 錯誤。

#### 3.4 網頁地圖技術

Web 端地圖採用 Leaflet 開源框架 [10]。其核心基於 Tile Layer 技術，將地圖切割為大量小圖塊以降低載入負擔。本系統利用 Leaflet 實現北台灣地圖呈現、HTML5 Geolocation API 即時定位，以及互動式景點標記 (Markers) 功能。

### 四、系統實作

#### 4.1 Web 端使用者互動介面

在介面設計上，我們使用 Gradio 打造了一個簡潔直觀的操作環境。



圖2:行程規劃選單介面



圖3:下載按鈕

行程下載：系統將 AI 生成的建議內容（包含景點介紹、交通方式、天氣提醒）格式化後，提供 .txt 文字檔下載功能，方便使用者離線閱讀。

當使用者詢問行程時，AI 的回覆結構清晰，首先提示當地氣溫與降雨機率，接著列出景點特色與門票資訊，最後提供交通轉乘建議。

#### 4.2 Android 行動端實作

為滿足旅途中的即時導覽需求，本專題使用 Android Studio 開發了行動應用程式，整合以下功能模組：

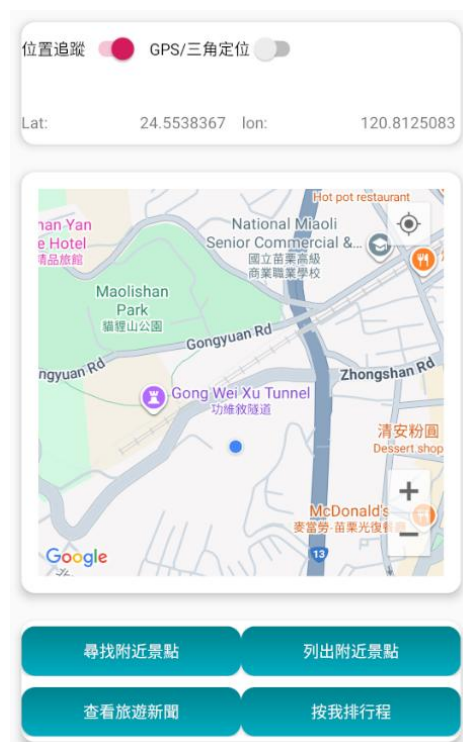


圖4:地圖介面



圖5:周邊景點搜尋 (Nearby Search)



圖6:旅遊新聞內文

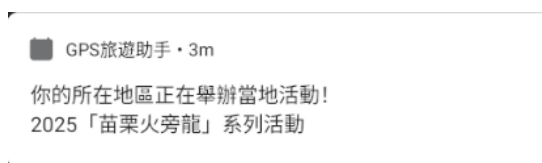


圖7:附近活動通知



圖8: LLM 助手無縫整合

## 五、結論與未來展望

### 5.1 結論

本專題成功開發了一套整合 LLM、RAG 技術與政府開放資料的「AI 旅遊通」系統。透過向量資料庫的知識增強，系統有效解決了通用 AI 模型在旅遊規劃時的幻覺問題；藉由 TDX 與 CWA API 的串接，賦予了 AI 掌握即時交通與天氣資訊的能力。此外，結合 Web 端與 Android 端的跨平台實作，為使用者提供了從「行前規劃」到「旅途中導覽」的完整解決方案。

### 5.2 未來展望

未來研究可朝以下方向延伸：

1. **擴充資料覆蓋範圍：** 目前主要針對北台灣景點，未來可擴充至全台資料庫。
2. **引入協同過濾推薦：** 結合使用者的評分與瀏覽紀錄，引入推薦系統演算法，提供更精準的景點推薦。
3. **多語言支援：** 利用 LLM 的翻譯能力，開發多語言介面，服務國際旅客。

## 六、專題作品影音平台網址

專題成果展示 - AI 旅遊通

[https://youtu.be/Lw4lg\\_ubumY](https://youtu.be/Lw4lg_ubumY)

## 七、參考文獻

### [中文文獻]

[1] 中央氣象署 (CWA), "氣象資料開放平台 API 使用手冊," [Online].

Available:

<https://opendata.cwa.gov.tw/>

[2] 交通部運輸資料流通服務 (TDX), "TDX API 開發者說明文件," [Online].

Available:

<https://tdx.transportdata.tw/>

[3] 交通部觀光署, "北台灣旅遊資料," [Online]. Available:

<https://www.taiwan.net.tw/> [

4] 台北市政府觀光局, "景點介紹," [Online]. Available:

<https://www.travel.taipei/>

[5] 苗栗文化觀光旅遊網, "苗栗縣政府文化觀光局," [Online]. Available:

<https://miaolitravel.net/>

[6] "Google Maps API 學習筆記 - 4: Place API 自動完成地址、地點評論摘要," *Let's Write*. [Online]. Available:

<https://www.letswrite.tw/google-map-api-place-api/>

[7] "快速建構地圖服務 (四) - 當 Leaflet JS 遇見 Data Hub," *iThome*, 2021. [Online]. Available:

<https://ithelp.ithome.com.tw/articles/10235399>

### [英文文獻]

[8] Bootstrap, "Bootstrap 5 official documentation," [Online]. Available:

<https://getbootstrap.com/docs/5.0/getting-started/introduction/>

[9] Hugging Face, "Text Generation Inference & Router API," [Online].

Available:

<https://huggingface.co/docs>

[10] Leaflet.js, "Leaflet.js official documentation," [Online].

Available: <https://leafletjs.com/>

[11] MDN Web Docs, "Web API references," [Online]. Available:

<https://developer.mozilla.org/>