1. 前言

1-1 背景介紹

隨著都市生活步調加快與外食文化盛行，餐飲選擇愈趨多樣，消費者面對資訊過載與選擇困難的情況愈發嚴重。現有平台多依賴靜態評分與排行榜，不僅缺乏即時互動性，也難以針對個人偏好進行精準推薦，導致使用者在面對眾多選項時，往往無法快速做出合適選擇。

1-2 動機

目前主流的美食推薦平台，仍停留在傳統搜尋與篩選階段，缺乏智能化與個性化的服務，無法即時理解使用者需求或根據語意進行互動回應。此外，多數評論僅呈現整體星等，難以具體反映「服務」、「氣氛」、「餐點品質」等細節。因此我們希望打造一套整合深度學習與語言模型的智慧推薦系統，提升用戶體驗與推薦品質。

1-3 系統目的與目標

本系統旨在打造一套能即時理解使用者需求並進行個性化美食推薦的智慧平台。具體目標包括：

* 建立自然語言互動式的推薦介面，提供即時問答與美食建議。
* 分析評論內容，細分服務、餐點、氣氛等面向，呈現餐廳真實表現。
* 建構使用者偏好學習機制，讓系統能隨使用次數自我優化推薦結果。
* 提供地圖導航與附近探索功能，整合使用者行動情境。

1-4 預期成果

* 提供即時、互動式美食推薦服務，有效減少使用者搜尋時間與選擇困難。
* 優化使用者體驗，讓推薦結果更貼近個人口味與需求。
* 提升平台評論資料價值，建構可信賴的資訊生態。
* 建立可擴充的智慧推薦架構，未來可應用至旅遊、娛樂等其他生活服務領域。

第**2**章營運計畫

2-1可行性分析

【技術可行性】

本系統整合大型語言模型（如 OpenAI ChatGPT）進行人機對話，透過自然語言處理（NLP）技術，理解使用者需求並提供人性化回應。推薦系統核心採用基於深度學習的神經網路架構，使用EfficientNet-B4 處理餐點圖片分類，以及使用 BERT 模型進行情感分析，解析評論中隱含的情緒與偏好。為了進一步優化推薦品質，導入協同過濾與強化學習機制，能根據使用者歷史與即時反饋進行動態調整與學習。系統也整合 Google Maps API 以即時查詢地點、營業資訊與距離等資料，提升推薦的時效性與實用性。整體架構模組化，可根據後續發展需求彈性擴充，例如未來可加入多語言模組（如 Google Translate API 或 DeepL）以支援國際化應用。

【市場可行性】

隨著外食人口的成長與旅遊需求的恢復，消費者對於快速、個人化的美食推薦工具需求越來越強烈，尤其在資訊爆炸的環境中，「不知道吃什麼」、「不知道該相信哪個評論」等問題已成為常見困擾。現有平台如 Google Maps 或 Yelp 雖然提供搜尋與篩選功能，但多以靜態清單或星級排序為主，缺乏深層理解與互動推薦的能力。與此相比，本系統的主要差異化優勢，在於能根據使用者即時對話與情境，提供動態且精準的餐廳建議。透過聊天的對話方式，消費者不需學習複雜操作，只要輸入自然語言就能獲得回應，使用習慣與現有 AI 助理高度接近，大幅降低使用門檻。目標市場涵蓋上班族、學生、旅客與外送需求者等，均為日常需快速決策的高使用族群。

【經濟可行性】

在經濟面評估上，系統開發與維運成本可控制且分階段投入。初期開發所需投入主要為語言模型整合、資料標註與前後端架構建置，並包含與 Google Maps API 的串接費用、伺服器與資料儲存等雲端資源支出。由於多數功能可依需求逐步擴充，初期可以僅針對特定地區或族群進行測試營運，控制預算風險。在收益模式方面，系統可透過導流廣告收費、與商家合作的推薦抽佣，以及提供高級會員制服務（如收藏清單、偏好設定記憶等）來創造穩定收入來源。長期而言，也可發展合作平台如觀光局、美食部落客等外部資源，創造異業加值效益。

2-2商業模式（Business Model）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 關鍵合作夥伴 | 關鍵活動 | 價值主張 | | 顧客關係 | 目標客群 |
| * 餐廳商家 * 部落客與KOL * 地圖導航服務提供商（如Google Place API） | * 系統開發與優化 * 數據蒐集與模型訓練 * 行銷推廣與品牌經營 * 用戶反饋 | **提供即時、個人化、情感分析加持的智慧美食推薦，幫助使用者快速選擇想吃的美食** | | * 聊天式互動體驗 * 自動學習個人偏好 | * 上班族 * 學生 * 旅行者 * 外食族 * 選擇障礙者 |
| **關鍵資源** | **通路** |
| * 餐廳資料庫 * LLM、NPL、情感分析模型等 * 推薦系統 * 開發技術人員 | * App Store、Google Play * 官方網站 * 社群媒體 * 合作餐廳曝光 |
| **成本結構** | | | **收益來源** | | |
| * 系統開發、維護成本、伺服器與資料儲存費用 * 行銷推廣支出 * 授權技術費用 | | | * 基本版免費 * 訂閱版收費（VIP功能，如無廣告、更深度推薦） * 廣告置入 * 餐廳合作推廣費 | | |

2-3 市場分析 – STP

【市場區隔（Segmentation）】

人口變數：

* 年齡：18-40歲為主（學生、上班族、年輕旅行者）。
* 地區：都市地區或外縣市旅行時。
* 職業：學生、上班族、旅遊者。

心理變數：

* 生活步調快，重視效率與便利有。
* 選擇障礙，容易在眾多餐廳中猶豫不決。
* 科技接受度中高，樂於使用科技產品與聊天互動。

行為變數：

* 外食頻率高，需要即時找到合適餐廳。
* 在旅行時尋找在地餐飲傾向依賴推薦或評論來快速決定餐廳。
* 喜歡或習慣使用即時互動式服務（如聊天機器人、推薦系統）。

【目標市場選擇（Targeting）】

選擇策略：差異化市場策略（針對不同場景，如日常外食、旅行需求，提供智慧即時推薦服務）

主要目標市場：忙碌的上班族與學生外縣市旅行者與短期出差者有選擇障礙、資訊焦慮、希望快速決策的外食族

【市場定位（Positioning）】

品牌定位：你的貼身智慧美食顧問，快速推薦、懂你所需

價值主張：結合深度學習與自然語言處理，提供即時、聊天式互動推薦，幫助使用者快速從海量資訊中找到最符合自己需求的美食選擇，無論是日常用餐還是外地旅行，都能即時解決選擇困難。

2-4SWOT-TOWS分析

|  |  |  |
| --- | --- | --- |
| 內部 / 外部因素 | 機會 (O)  (1) 個性化需求增：用戶對標準化評分系統不滿足。  (2) AI 技術進步：提升推薦準確度與效率。  (3)旅遊市場成長：疫情後旅遊與美食需求上升。 | 威脅 (T)  (1)競爭對手強：Google、Yelp 擁有龐大數據與用戶。  (2) 數據隱私挑戰：需符合 GDPR、CCPA 等法規。  (3)維運成本高：AI 運行與維護費用大。 |
| 優勢 (S)  (1)個性化推薦：AI 根據用戶喜好動態推薦。 (2)即時互動：對話式介面提升用戶體驗。 (3)多語言支持：解決語言障礙，拓展市場。  (4)地理資訊整合：GPS+導航，提供即時推薦。 | SO 策略  (1)市場滲透：利用 AI 個性化推薦，吸引自由行旅客與商務旅客。 | ST 策略  (1)技術護城河：不斷優化 AI 模型，提高推薦準確度，增強競爭優勢。 (2)數據隱私強化：建立安全機制，確保用戶數據安全，符合 GDPR、CCPA 等法規。 (3)品牌差異化：強調「即時互動 + 動態推薦」，區別於 Google Maps、Yelp。 |
| 劣勢 (W)  (1)技術門檻高：需整合 AI、NLP、GIS 等技術。 (2) 數據依賴外部：須與 Google Maps 等合作，受限權限。 (3) 初期用戶少：數據不足，推薦效果受影響。 | WO 策略  (1)數據收集與優化：透過用戶回饋機制，改善冷啟動問題，提高 AI 推薦品質。 | WT 策略  (1)降低 API 依賴：發展自有餐廳數據庫，減少對 Google Maps 的依賴。 (2)多平台發展：不僅限於旅遊市場，也可應用於本地居民的美食推薦。 (3)加強用戶教育：透過部落格、社群推廣，提升 AI 美食推薦的信任度。 |

第**3**章系統架構

3-1 系統架構

本系統功能分成使用者及管理者：

使用者功能

1.智能美食推薦

* 進入首頁輸入「想吃火鍋」、「適合聚餐的日式餐廳」等語句。
* 系統即時回應推薦結果，包含餐廳資訊、評價與距離。
* 點擊餐廳可查看詳細介紹與評論。

2️.個人偏好設定

* 使用者在初次使用或設定頁面可填寫口味、預算、地點等偏好。
* 日後推薦會自動依據歷史使用行為進行個人化調整。
* 可隨時編輯偏好選項。

3️.餐廳評論查閱與參考

* 進入餐廳頁面後可查看其他用戶的評價與圖片。
* 評價區分為服務、環境、餐點等分類，便於理解餐廳特點。
* 提供代表性評論摘要，快速掌握重點意見。

4️.餐廳搜尋與附近探索

* 點選「附近推薦」即可根據目前位置列出熱門餐廳。
* 可篩選餐廳類型、營業時間、適合人數等條件。
* 地圖模式查看各餐廳位置，點擊即可導航。

5️.餐後評論與互動

* 用完餐後可點選「寫下你的評論」，填寫星等與心得。
* 可上傳食物或店內照片，與其他用戶交流體驗。
* 自己的評論可隨時修改或刪除。

管理者功能

1.評論與內容管理

* 登入後可查看平台上所有新評論。
* 可針對不當內容進行審核、隱藏或刪除。
* 管理餐廳資料與圖片，確保資訊正確性。

2.使用者管理模組

* 查看使用者帳號狀態與互動紀錄。
* 可發送系統公告或限定活動通知。
* 管理黑名單或違規帳號。

3️.行銷與推廣活動管理

* 設定首頁廣告位、主打推薦餐廳。
* 發布期間限定優惠、主題餐廳推薦等行銷活動。
* 可查看行銷成效報表，了解活動點擊與參與情形。

4.系統報表與效能追蹤

* 查看每日活躍使用者數、熱門搜尋、推薦點擊率等數據。
* 根據使用數據微調推薦邏輯與餐廳展示策略。
* 匯出報表供團隊內部分析與決策。

3-2 系統軟、硬體需求與技術平台

表 3-2-1 系統軟、硬體需求

|  |  |
| --- | --- |
| 硬體需求 | |
| 作業系統 | Windows、Mac |
| 裝置需求 | |
| 手機網路需求 | WiFi/行動數據 |
| 手機定位需求 | 能使用GPS定位 |

3-3 開發標準與使用工具

表 3-3-1 開發工具

|  |  |  |
| --- | --- | --- |
| 系統開發環境 | | |
| 作業系統 |  | Windows |
| 撰寫工具 |  | Visual Studio Code |
| 程式開發語言 | | |
| 前端 |  | HTML、JS、CSS |
| 後端 |  | python |
| 資料庫 |  |  |
| 文件及美化工具 | | |
| 文件 |  | Microsoft Word |
| 簡報 |  | Microsoft PowerPoint、Canva |
| 專案管理及版本控制工具 | | |
| 應用程式 |  | Fork |
| 版本控制 |  | Git |
| 開發輔助工具 | | |
| AI工具 |  | ChatGPT |

第**4**章專案時程與組織分工

4-1專案時程：甘特圖

表 4-4-1 甘特圖

預期進度

實際進度

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 113年 | | 114年 | | | | | | | | | | |
| 任務  描述 | 11月 | 12月 | 1月 | 2月 | 3月 | 4月 | 5月 | 6月 | 7月 | 8月 | 9月 | 10月 | 11月 |
| 主題 思考 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 功能 討論 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 介面 設計 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 前端 網頁 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 後端 開發 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 系統 整合 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| AI 對話 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 編寫系統手冊 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 系統功能測試 |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

4-2專案組織與分工

表 4-2-1 專案組織與分工表

●主要負責人 〇次要負責人 (每一項只能有1位主要負責人，次要負責人最多2位)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 項目/組員 | | 11146070/  歐祐詮 | 11146078/  王語彤 | 11146080/  李啓慧 | 11146084/  林愉晴 |
| 後端開發 | 資料庫建置 |  |  |  |  |
| 伺服器架設 |  |  |  |  |
| 功能A |  |  |  |  |
| 功能B |  |  |  |  |
| 功能C |  |  |  |  |
| 前端開發 | Template A |  |  |  |  |
| Template B |  |  |  |  |
| Template C |  |  |  |  |
| Template D |  |  |  |  |
| 美術設計 | UI/ UX |  |  |  |  |
| Web/APP介面設計 |  |  |  |  |
| 色彩設計 |  |  |  |  |
| Logo設計 |  |  |  |  |
| 素材設計 |  |  |  |  |
| TEST A |  |  |  |  |
| 文件撰寫 | 統整 |  |  |  |  |
| 第1章 前言 |  |  |  |  |
| 第2章 營運計畫 |  |  |  |  |
| 第3章 系統規格 |  |  |  |  |
| 第4章 專題時程與組織分工 |  |  |  |  |
| 第5章 需求模型 | 提醒，  第6、第7章各部制模型不同，記得刪除不需要的模型名稱 |  |  |  |
| 第6章 程序或設計模型 |  |  |  |  |
| 第7章 資料或實作模型 |  |  |  |  |
| 第8章 資料庫設計 |  |  |  |  |
| 第9章 程式 |  |  |  |  |
| 第10章 測試模型 |  |  |  |  |
| 第11章 操作手冊 |  |  |  |  |
| 第12章 使用手冊 |  |  |  |  |
| 報告 | 簡報製作 |  |  |  |  |

註：後端開發、前端開發及美術設計視各組專題功能新增項目，文件撰寫及報告則應固定欄位

專題成果工作內容與貢獻度表

本組成員之工作內容與貢獻度**(**加總為**100%)**，請依組員人數自行調整欄位

|  |  |  |  |
| --- | --- | --- | --- |
| 序號 | 姓名 | 工作內容<各限100字以內> | 貢獻度 |
| 1 | 歐祐詮 |  | % |
| 2 | 王語彤 |  | % |
| 3 | 李啓慧 |  | % |
| 4 | 林愉晴 |  | % |
|  | | | 總計:100% |

4-3上傳GitHub紀錄

第**5**章需求模型

5-1使用者需求

* 功能性需求

表 5-1-1功能性需求表

|  |  |
| --- | --- |
| 項目 | 說明 |
| 用戶註冊和登入 | 允許用戶創建帳號，並透過Gmail或簡單的帳密登入系統。 |
| 互動式對話推薦美食 | 利用ChatGPT進行互動式對話，收集旅行者的飲食偏好（如喜歡的菜系、偏好口味、價格範圍）。 |
| 動態導航與及時定位推薦 | 整合 GPS 或 Google Maps API，實現即時推薦功能和顯示各地方餐廳。 |
| 用戶反饋 | 設計簡單的評分和評論系統，讓用戶在使用推薦後可以留下回饋。 |
| 分析使用者偏好 | 提取關鍵詞， 將關鍵詞與美食數據庫中的餐廳信息進行匹配。 |
| 生成推薦清單 | 基於匹配結果生成一份排序的餐廳推薦清單，並提供詳細介紹（如餐廳地址、菜單特色、營業時間）。 |
| 推薦結果過濾與排序 | 增加用戶自定義過濾功能，例如按餐廳評分、距離、價格範圍、菜系等條件篩選結果。 |
| 個人資料管理 | 用戶可以自行修改頭像、姓名等個人資訊。 |

* 非功能性需求

1.安全性：系統需妥善保護使用者資料，所有個人資訊皆僅儲存在本地裝置，並採取適當措施防止資料外洩或未經授權的存取。

2.可用性：使用者界面需簡單易懂，讓使用者無需操作說明可完成推薦流程，並支援多語言操作與顯示。

3.可靠性：系統需保持穩定運作，確保主要功能在正常條件下可持續執行，並將系統故障率控制在可接受範圍內。

4.可維護性：系統設計需具備良好的可讀性與模組化結構，便於後續進行錯誤修正、功能更新或系統優化，以提升維護效率。

5.可擴展性：系統架構需具備良好的擴充能力，便於未來新增功能、支援更多使用者或整合其他外部服務時，能以最小代價進行擴展與修改。

6.相容性：前端介面需支援主流瀏覽器（如 Chrome、Firefox、Safari）以及行動裝置（iOS、Android）之正常顯示與操作。同時，Google Maps、ChatGPT API需正確整合，並妥善處理各平台之相容性問題，確保功能穩定運作。