

國立雲林科技大學工業工程與管理系

碩士論文

Department of Industrial Engineering and Management

National Yunlin University of Science & Technology

Master Thesis

利用生成對抗網路實現旅遊推薦系統

Using Generative Adversarial Networks to Implement

Travel Recommendation Systems

邱冠博

Kuan-Po Chiu

指導教授：蘇純繒 博士

Advisor：Chun-Zeng Su, Ph.D.

共同指導教授：陳奕中 博士

Co-Advisor：Yi-Chung Chen, Ph.D.

中華民國 112 年 6 月

June 2023

國立雲林科技大學
碩士班學位論文考試委員會審定書
National Yunlin University of Science and Technology
Thesis Oral Defense Approval Form

本論文係 邱冠博 君在本校 工業工程與管理系碩士班 所提論文 利用生成對抗網路實現旅遊推薦系統 碩士資格水準，業經本委員會評審認可，特此證明。

The student CHIU.KUAN-PO enrolled in the Master's program in Department of Industrial Engineering and Management has satisfactorily passed the oral defense of the thesis Using Generative Adversarial Networks to Implement Travel Recommendations Systems.

口試委員：
Oral defense
committee members

蘇純繪

蘇純繪

陳奕中

陳奕中

洪國鈞

洪國鈞

陳維婷

陳維婷

指導教授：
Advisor(s)

蘇純繪

蘇純繪

陳奕中

陳奕中

所長：
Dean of Graduate
Institute

傅家啓

中 華 民 國 112 年 3 月 22 日

摘要

隨著旅遊產業的蓬勃發展，旅行門檻逐年降低，使得選擇自由行的旅客日益增多，然而在安排行程時總是會花上大量的時間與精力，因此許多旅遊平台引入了旅遊推薦系統，試圖降低旅客安排行程的時間。然而，過往旅遊產業上的推薦系統容易遇到兩大問題(1)用戶過往資料不足或旅遊頻率太低所造成的持續性冷啟動、(2)景點存在某些無法量化的資訊。因此本研究針對過往旅遊推薦系統問題提出全新的框架，(1)為搜集過往旅客分享的遊記，如許多遊客所撰寫有關東京的遊記，讓推薦系統學習當下地區該如何安排行程。(2)利用生成對抗網路(Generative Adversarial Network, GAN)，使用過往旅客分享的遊記，產生擬真的行程。最終本研究將透過嘗試不同模型框架以及不同的懲罰函數，目標為建立一個能夠產生與真實行程差異較小的生成對抗網路。

關鍵字：旅遊、旅遊推薦系統、生成對抗網路



ABSTRACT

Rapid development of the travel industry continues to lower barriers to independent travel. However, arranging an itinerary remains time-consuming; many online travel platforms have thus introduced travel recommendation systems. Existing travel recommendation systems are prone to two major problems: (1) persistent cold starts caused by insufficient user data or low frequency of travel, and (2) the existence of certain unquantifiable information about tourist attractions. Therefore, this study proposes a novel framework with the following contributions: (1) the collection of travelogues shared by travelers as data for itinerary suggestions and (2) application of the generative adversarial network (GAN) to this data for the generate of realistic itineraries. The proposed system is demonstrated using existing data and trying different modeling frameworks and different penalty functions, with the goal of building a generative adversarial network that can produce a real itinerary with less variance from the real one.

Keywords : Tourism, Travel Recommendation, Generative Adversarial Networks

誌謝

轉眼兩年的研究生生涯即將結束，這也將會是本人最後的求學階段，代表自己即將結束校園生活，真正的踏入社會，心中自然免不了有些許失落，但更多的是對未來的期待，也感謝在這段時間遇到熱心助人的學長姊、師長以及同學，這篇論文能夠完成，他們功不可沒。首先要感謝我的指導教授蘇純繒老師以及陳奕中老師，純繒老師總是能在我的研究當中挑出重要的地方進行評論，其中對於論文的想法也是源源不絕，深感敬佩。也許我怎麼都不會想到，當初在大學面試階段，在其他學校的演講台上看到的老師，竟會成為我大學的班級導師，更是在日後成為了我的指導教授以及我的好友，我想這就是用資料科學也解釋不清楚的「緣分」吧，非常感謝奕中老師讓我在工管這一領域能夠尋找到自己的興趣，帶領我認識資料科學、大數據，鼓勵我們參加研討會，更是鼓勵我前往日本進行研究交換。可以說，沒有奕中，就沒有今天的我。也謝謝實驗室博士班學長姐們，謝謝東綺學長、映均學姊還有志豪學長，時常跟我分享一些關於這個領域的趣事，也能夠在不論是大學部專題報告還是研究所論文報告上給予相當犀利的意見，促進我思考以及成長，「任何事情都要有自己的想法」這是他們帶給我最寶貴的禮物。感謝富正大哥，在交流的過程中總是能夠提出與眾不同的意見與想法，改變了我的思考模式。

特別感謝亮岑，若是沒有她，我今天將沒有辦法前往日本研究交換，也沒有辦法加入奕中老師的實驗室進行專題研究。也感謝我的實驗室同學承安、長宏、堉豪以及智偉，不論是在分工還是平常的相處，我們總是能夠相處得很融洽，又能夠在該認真的時候認真，謝謝你們。感謝韋誼、羿文、崇瑋、鈺傑、添翼以及元緻，謝謝你們讓實驗室總是充滿著笑聲。最後感謝我的家人們，沒有他們在這趟求學路途上的支持與鼓勵，我也不可能走到現在這一步。

最後，謹此向所有幫助過我的人致上最大的敬意與謝意，這一篇論文的成果，獻給你們。

目錄

摘要.....	i
ABSTRACT.....	iii
誌謝.....	iv
目錄.....	v
表目錄.....	vii
圖目錄.....	viii
第一章 緒論.....	1
1.1 研究背景與動機.....	1
1.2 研究目的.....	3
1.3 研究範圍.....	4
1.4 研究限制.....	4
1.5 研究流程.....	5
第二章 文獻回顧.....	7
2.1 ChatGPT 應用於文字探勘.....	7
2.2 旅遊推薦系統.....	8
2.2.1 以使用者為主的旅遊推薦系統.....	8
2.2.2 以景點為主的旅遊推薦系統.....	8
2.2.3 混合型旅遊推薦系統.....	8
2.3 生成對抗網路(Generative Adversarial Network).....	9
第三章 資料集.....	10
3.1 遊記資料集.....	10
3.2 景點資料集.....	11
第四章 研究方法.....	13
4.1 透過 ChatGPT 進行景點順序提取.....	13

4.1.1	文本處理.....	13
4.1.2	ChatGPT 景點順序提取	14
4.2	實際行程網格資料.....	14
4.3	景點條件網格資料.....	15
4.4	建立 GAN 模型	16
4.4.1	GAN 架構.....	16
4.4.2	全連接層架構.....	18
4.4.3	卷積層架構.....	18
4.4.4	殘差網路與最大池化架構.....	19
4.4.5	損失函數.....	20
4.5	模型評估.....	20
第五章	實驗模擬.....	22
5.1	清洗過程.....	22
5.1.1	遊記文章清洗.....	22
5.1.2	ChatGPT 景點順序提取	23
5.1.3	景點爬蟲.....	25
5.1.4	篩選區域過程.....	26
5.2	建立網格.....	28
5.3	不同架構的 GAN 對預測結果之影響.....	28
5.3.1	全連接架構.....	28
5.3.2	卷積網路架構.....	30
5.3.3	殘差網路與最大池化架構.....	33
5.4	績效探討.....	34
第六章	結論與未來研究.....	37
參考文獻	38

表目錄

表 1 遊記資料集.....	10
表 2 景點資料集.....	11
表 3 文章處理前後比較表.....	14
表 4 ChatGPT 輸入以及輸出	14
表 5 景點組合表.....	16
表 6 文章標題含有「推薦」的文章內容.....	22
表 7 文章標題包含「迪士尼」的文章內容.....	23
表 8 文章天數詞提出.....	23
表 9 文章 Token 計算.....	24
表 10 ChatGPT 完整輸入及輸出表	24
表 11 五種類型輸入準確率比較表.....	35
表 12 改良 MAE 績效比較表.....	35



圖目錄

圖 1 國外旅遊行為調查結果.....	1
圖 2 旅行酒吧網站示意圖.....	2
圖 3 研究目的流程圖.....	3
圖 4 PTT 日旅版畫面	4
圖 5 研究流程圖.....	6
圖 6 GAN 架構圖	9
圖 7 PTT 旅遊網誌圖	10
圖 8 每季文章數量直方圖.....	11
圖 9 景點分佈圖.....	12
圖 10 研究流程圖.....	13
圖 11 實際行程網格示意圖.....	15
圖 12 景點條件網格示意圖.....	16
圖 13 本研究 GAN 架構圖.....	17
圖 14 擬真行程網格.....	17
圖 15 生成器全連接層架構.....	18
圖 16 生成器卷積層架構.....	19
圖 17 生成器殘差網路架構.....	19
圖 18 判別器最大池化架構.....	20
圖 19 改良 MAE 計算示意圖.....	21
圖 20 爬蟲步驟示意圖.....	25
圖 21 遊記中出現過所有景點分布圖.....	26
圖 22 旅遊型態長條圖.....	26
圖 23 行程景點數量盒鬚圖.....	27
圖 24 東京都內旅遊景點分布圖.....	27

圖 25 東京 23 區景點分布圖.....	27
圖 26 單一損失函數全連接層第 6000 Epoch 訓練結果	29
圖 27 單一損失函數全連接層第 7000 Epoch 訓練結果	29
圖 28 單一損失函數全連接網路 Loss 曲線	29
圖 29 三種損失函數全連接層第 7000 Epoch 訓練結果	30
圖 30 三種損失函數全連接層 Loss 曲線	30
圖 31 單一損失函數卷積網路第 6000 Epoch 訓練結果	31
圖 32 單一損失函數卷積網路第 7000 Epoch 訓練結果	31
圖 33 單一損失函數卷積網路 Loss 曲線	32
圖 34 三種損失函數卷積網路第 6000 Epoch 訓練結果	32
圖 35 三種損失函數卷積網路第 7000 Epoch 訓練結果	32
圖 36 三種損失函數卷積網路 Loss 曲線	33
圖 37 殘差網路與最大池化第 6000 Epoch 訓練結果	33
圖 38 殘差網路與最大池化第 7000 Epoch 訓練結果	34
圖 39 殘差網路與最大池化 Loss 曲線	34
圖 40 殘差網路與最大池化測試資料預測結果	35
圖 41 殘差網路與最大池化測試資料預測結果	36

第一章 緒論

1.1 研究背景與動機

隨著經濟發展、物質生活提升的社會下，許多人會選擇外出旅遊作為一種放鬆享受的方式，舉凡是全家一起出門遊玩的家族旅遊、公司為了犒賞員工而舉辦的員工旅遊等都是非常常見的，旅遊業也因此而快速發展。根據根據世界經濟論壇發布的 2019 年度「旅遊及觀光競爭力報告」中指出，旅遊業佔了全球 GDP 的 10%，而旅行門檻的降低後，讓民眾在旅遊的這個層面需求也是日益增加，除了在自己國家的景點觀光遊玩，出國也是一個熱門的選項。據台灣觀光年報統計 2019 年國人出國人次共計 1710 萬[21]，相較於 2018 年增加了約 50 萬的人次，而目的地多以亞洲地區的日本為主，有將近 500 萬的人次。但相較於讓旅行社為旅客規劃好所有的行程的跟團旅行，近年因為深度旅遊的概念崛起，比起可以看完所有推薦景點的那種經濟實惠類型的緊湊行程，民眾較為偏好對於一個景點的深度觀光，因此許多旅客逐漸偏好採用自助旅遊的方式出國旅行。根據台灣趨勢研究於 2019 發布的《國外旅遊行為調查》[23]表示，近期三年內的統計，約有 75% 的旅客在出國旅遊時選擇自助旅遊，如圖 1 所示。調查中也指出，旅客對於自助旅行最感到困惑的其中一個地方為「行程規劃與路徑選擇」，顯示出自助旅行常需要為了此次旅遊花費大量的時間進行目的地的研究及行程的安排。

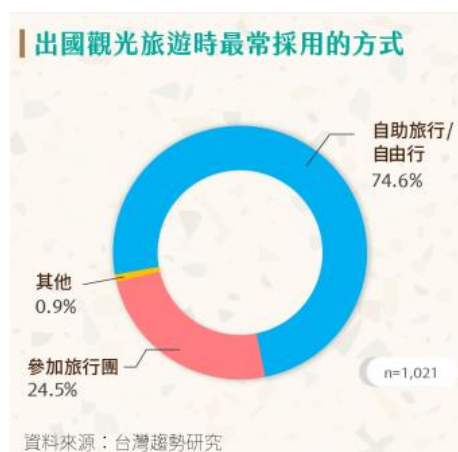


圖 1 國外旅遊行為調查結果

在資訊化時代及人工智慧的發展，推薦系統逐漸被應用在旅遊產業上，藉此幫助改善使用者在選擇景點上所花的時間，許多不同的旅遊推薦產品開始被推出，例

如:Funliday、旅行酒吧等，分別可以根據使用者的定位進行適合景點推薦或是推薦旅行地的特色景點、飯店、餐廳，示意圖如圖 2 所示。舉例來說，若兩位使用者 A 與使用者 C 的旅遊類型、年齡、職業是相近的，旅遊推薦系統便會推薦使用者 A，使用者 C 曾經遊玩過的景點或是行程，或是景點特徵為主，例如:使用者 A 較喜歡到名勝古蹟做觀光，旅遊推薦系統便會推薦像是雷門等類似的景點給使用者 A 做為參考。

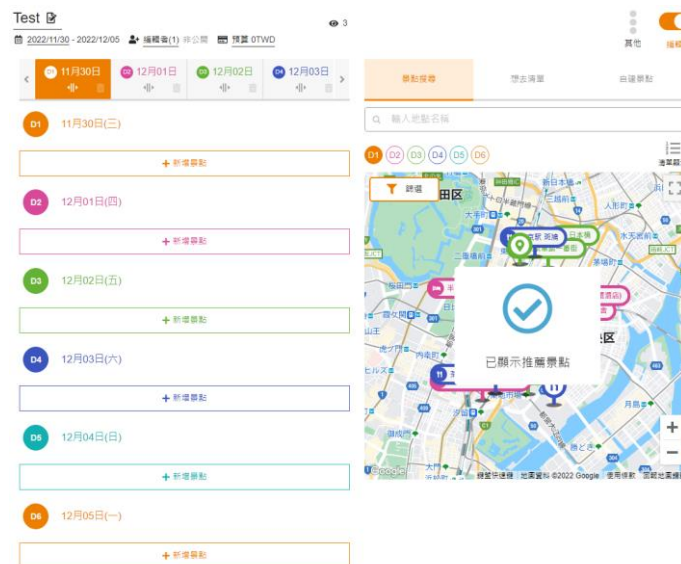


圖 2 旅行酒吧網站示意圖

而在學術界中，旅遊推薦系統的實現方式主要分為利用**協同過濾(Collaborative Filtering)**或者是透過**機器學習**等方式來進行推薦，而在目標上則有通用的推薦系統，或者是個人的推薦系統；在 2018 年 Coelho 等人[4]使用機器學習建立一個分類模型來自動識別用戶在 twitter 上的旅遊文章，並將其作識別為用戶的興趣點(Points of Interest, POI)，為了能夠更好的使模型客製化，用戶的好友以及追隨者的貼文也會一併考慮進模型內，最終在自願者的協助底下，達到了將近七成的準確率。在 2009 年 Subramaniaswamy 等人[18]透過一些社群網站上一些打卡的照片，並識別照片當中所在地的地理位置標籤，之後蒐集用戶的相關訊息例如：性別、年齡、旅遊季節，除此之外還會透過相關訊息來判斷旅遊群體的類別；之後使用貝葉斯學習模型預測用戶可能會喜歡的地點實現旅遊推薦系統。2011 年 Fenza 等人[5]提出基於協同過濾的混和型上下文感知系統，其目標為能夠根據用戶的個人資料以及情境，自適應的主動推薦用戶興趣點，也能夠稱之為情境感知推薦系統，此種類型系統很好的處理了不確定用戶興趣點的問題。

綜合過去相關研究，本研究發現旅遊推薦系統一直存在兩大問題分別是 (1)冷啟動問題及 (2)部分資訊難以量化問題。首先是冷啟動問題，以下分別用兩種情形舉例說明：情境一，用戶本身並沒有旅遊經驗，因此用戶無法確定自己真正想旅遊的景點類別，而導致系統也因為沒有過去的歷史資料可以進行比對而無法推薦給用戶適合他的旅遊行程。情形二，雖然用戶擁有非常豐富的旅遊經驗，但是因為旅行的目的地每次都不相同，所以過往的歷史旅行資料不一定適用於下一趟旅行，上述兩個情形反映出的問題就是因為資訊量過少導致的冷啟動問題。至於第二個問題部分資訊無法量化則是因為過往的旅遊推薦系統雖然有考慮到景點的開放時間、景點之間的距離以及交通時間等可以量化的數據，但卻無法考量到用戶「為何」選擇在某個時間前往某個景點。例如景點A開放時間是從早上十點開放至晚上十點且類別為購物，但該景點A卻是以夜景著稱，這些無法量化但是會影響遊客是否前往或前往時間的資訊，在以往的旅遊推薦系統中是無法被考量進去的。

1.2 研究目的

本研究提出了兩大概念來克服上述問題，(1)為搜集過往旅客分享的遊記，如：許多遊客所撰寫有關東京的遊記，讓推薦系統學習當下地區該如何安排行程，(2)利用生成對抗網路(Generative Adversarial Network, GAN)與概念(1)中的搜集過往旅客分享的遊記克服資訊無法量化的問題，是因為在旅客分享的遊記當中就已經包含景點的選擇原因如：時間、交通等，而生成對抗網路可以在學習原因後，生成與真實行程相似的擬真行程，藉此推薦用戶行程，其目的流程圖如圖 3 所示。

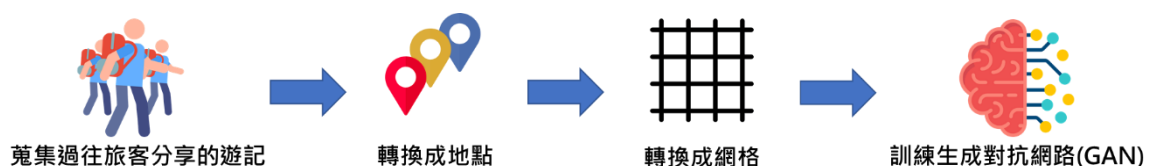


圖 3 研究目的流程圖

1.3 研究範圍

本論文將以日本東京地區為基礎進行我們的研究，因為過往統計資料顯示目前台灣人出國以日本為大宗，而根據日本國土交通省「平成 30 年宿泊旅行統計調查」中顯示，台人旅日較常選擇東京地區的比例為 16%，位居第一[22]。

本研究所蒐集的過往遊記來源為「PTT 實業坊」中的「日旅版」，PTT 實業坊為一個台灣電子佈告欄，以學術性質為原始目的，提供線上言論空間的平台，依照不同種類分成不同的看板，「日旅版」就是其中之一，日旅版畫面如圖 4 所示。



圖 4 PTT 日旅版畫面

本研究在遊記搜集的過程中也發現，在與日本旅遊有關的討論區中，有近 15%的文章皆為分享東京的遊記，故本研究的推薦系統主要以東京遊記為主。

1.4 研究限制

本研究的研究限制有以下五點：

1. 不考慮住宿地點的問題

由於搜集之資料集並無包含住宿地點，且住宿地點通常會等到旅遊行程安排結束後再進行決定，因此本次研究不考慮住宿地點。

2. 模型一次只能生產一天的行程

遊記大多數都是依照天數進行撰寫，且旅客在安排行程時較常思考第幾天要去哪裡等問題，因此本研究認為模型一次生產一天的行程最能符合使用者需求。

3. 完全相信遊記裡提及的內容

網路上所蒐集到的遊記，因要判別該遊記所提及的行程是否可行將耗費大量人力，故本研究將完全相信遊記裡的內容。

4. 全新景點不可能出現在遊記資料上，故不考慮

全新的景點因還沒有人去過，因此不會出現在本研究所搜集的遊記資料集中，因此本研究考慮的景點皆為目前已知的景點。

5. 景點提及次數不定 不考慮景點次數不平衡問題

某部分的景點被提及的次數比較少，在過往的統計方法中會造成資料不平衡的問題，但本研究使用生成對抗網路進行訓練，其資料不平衡的問題將對模型不會造成太大的影響。

1.5 研究流程

本研究的流程圖如圖 5 所示，以下為各部分說明：

1. 定義研究問題

提出過往旅遊推薦系統存在的缺點，利用過往旅客分享的遊記以及生成對抗網路就能夠在不需要用戶過往資料的情況下，產生出已經考慮景點無法量化資訊的行程，實現旅遊推薦系統。

2. 相關文獻回顧

針對 ChatGPT 進行文字探勘、旅遊推薦系統與生成對抗網路的過往研究進行探討，並介紹其中常見的幾個方法。

3. 資料集介紹與清洗

本研究搜集的遊記資料集為批踢踢日旅版中有提及「遊記」、「東京」的遊記資料，期時間範圍從 2015-2022 年。景點資料集則使用 Google Map 上搜尋「東京」所顯示出來的所位於東京都內的景點，後續清洗資料時會將不在原始景點資料即當中的景點回饋至景點資料即當中。

4. 建立模型

首先將遊記資料進行景點提取，確認好景點順序後轉換成本研究所需格式，後輸入 GAN 讓模型生成擬真行程資料

5. 模擬測試

本研究將遵循研究方法的架構將 GAN 中訓練好的生成器實際拿來使用，觀察其所生成的行程資料是否能符合使用者期望。

6. 結論

依據模擬結果驗證本研究所提出的概念是否能夠有效的解決上述問題，最終依照研究結果提出未來研究建議以及結論。

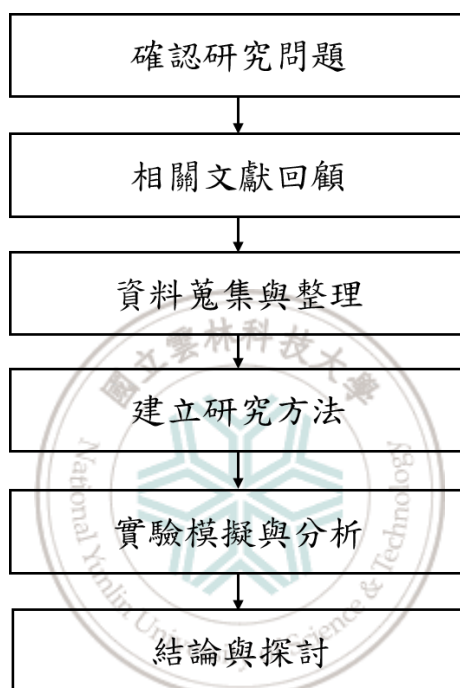


圖 5 研究流程圖

第二章 文獻回顧

本章將分為三個部分進行介紹，分別是 ChatGPT 應用於文字探勘、旅遊推薦系統以及生成對抗網路之相關研究。

2.1 ChatGPT 應用於文字探勘

人工智慧公司「OpenAI」在 2022 年 11 月 30 日發布了建立在 GPT-3(Generative Pre-trained Transformer 3) 上的聊天機器人程式 ChatGPT(Chat Generative Pre-trained Transformer)。GPT-3 是一種自回歸語言模型，目的為透過深度學習生成人類可以理解的自然語言(Natural Language)[2]，為有史以來參數最多的神經網路模型，包含了 1750 億個參數，而 GPT-3 在翻譯、問答、填空等問題展示出了驚人的成效，也表現出了零樣本(zero shot)與少樣本任務情況下的強大，近年來也有許多學者利用 GPT-3 在其他領域的自然語言問題(Natural Language Problem, NLP)上。通過廣泛的實驗結果和分析，展示了 ChatGPT 在不同類型的 NLP 任務中的有效性和當前的局限性。ChatGPT 在推理和對話任務方面表現良好；但同時，在解決特定任務，如序列標注等方面，ChatGPT 仍面臨挑戰。Qin 等人[16]於 2023 年通過廣泛的實驗結果和分析，展示了 ChatGPT 在不同類型的 NLP 任務中的有效性和當前的局限性。ChatGPT 在推理和對話任務方面表現良好；但同時，在解決特定任務，如序列標注等方面，ChatGPT 仍面臨挑戰。Aydın & Karaarslan[1] 於 2022 年發布首篇探討 ChatGPT 在文獻回顧章節上的應用，透過搜尋相關論文提供給 ChatGPT 進行改寫，爾後進行提問，提問的結果是好的，但經過比對後發現 ChatGPT 改寫大多都來自原始文章，由此可知其無法在原有的文章基礎上新增文字。Goyal 等人[7]於 2023 年比較了 GPT-3 與使用大量概要文本訓練好的模型，發現人們較為偏好 GPT-3 所生成的概要，而 GPT-3 還可以避免缺乏真實性等問題，同時探討如何提升 GPT-3 的結果。Jeblick 等人[11]於 2022 年利用 ChatGPT 簡化放射學報告，使其變得淺顯易懂，爾後經由 15 位放射學領域的專業人士進行評估，發現經由 ChatGPT 簡化過後在正確性與完整性上專家們給出一致好評，而在潛在傷害部分給出了褒貶不一的評

價。Tanner 等人[15]於 2022 年透過 GPT-3 分析學生們在線上課程的聊天室，經由少量的資訊來總結學生們在上課時所遭遇的問題，以幫助老師更有效的解決問題。

2.2 旅遊推薦系統

旅遊推薦系統依照其所使用到的資料大致上可分成三種：以使用者為主的旅遊推薦系統、以景點為主的旅遊推薦系統以及混合型旅遊推薦系統。

2.2.1 以使用者為主的旅遊推薦系統

此類型的推薦系統會依照用戶與其他使用者之間的相似程度來進行景點或行程上的推薦，例如小明曾經去過東京晴空塔、淺草雷門寺，使用者 A 曾經去過東京晴空塔、淺草雷門寺、上野公園，而在系統中發現，小明與使用者 A 的個人特徵相似，因此系統會推薦小明前往上野公園。Nitu 等人[14]於 2021 年分析用戶在推特上的推文，及其好友與追隨者的貼文，利用近因效應來過濾旅遊貼文實現旅遊推薦系統。Jia 等人[12]於 2015 透過量化用戶與其他使用者造訪過的景點與個人資料，計算彼此之間的餘弦相似度，將相似度較高的其他使用者造訪過的景點當作推薦依據。He 等人[8]於 2015，計算用戶之間的相似度、路線評級，使用 LDA 擷取出遊記裡的主題，利用相似度、評級以及遊記主題實現旅遊推薦系統。

2.2.2 以景點為主的旅遊推薦系統

此類型的推薦系統會依照景點之間的分數進行單一景點推，例如：小明曾經去過東京晴空塔、淺草雷門寺，爾後發現東照宮與淺草雷門寺相似程度頗高，則此類型推薦系統將會推薦小明前往東照宮遊玩。但用戶過往的旅遊經歷或評價比較難取得，且有可能會面臨資料不足的問題，因此大多數文獻經常與上述以使用者為主的類型混合使用。

2.2.3 混合型旅遊推薦系統

此小節展示了結合上述兩種類型為基底，並加上其他機器學習或方法實現旅遊推薦系統之過往研究。Huang & Bian[9] 於 2007 年利用貝葉斯網路合併上述兩種不同的推薦方法來實現旅遊景點推薦。Lu 等人[13]於 2012 年提出利用多重限制式的方式，量化所有可能的數值，且考慮到時間、路徑與交通時間等，並同時計算景點分類對於使用者的喜好程度。Ishanka & Yukawa[10]於 2018 年利用普拉奇克的情緒分析與使用者個性建制

旅遊推薦系統。Yu 等人[20]於 2017 年搜集行動定位服務資料，將其視為使用者造放果的景點，之後透過協同過濾以此景點進行推薦。

2.3 生成對抗網路(Generative Adversarial Network)

生成對抗網路最早在 2014 被 Goodfellow 等人[6]提出，屬於一種非監督式學習模型，其主要由兩個模型所構成，分別為生成模型與判別模型，生成模型的目的是要產生出能夠騙過判別器的假資料；判別模型的目的是要判別生成模型所產生的資料是否為真，並回傳結果至生成模型調整權重，經過不斷重複上述過程，直到判別模型無法辨別哪個才是生成模型所產生的資料，其架構如下圖 6 所示。近期也出現了許多改良 GAN 來產生圖片的研究。Reed 等人[17]於 2016 年提出 GAWWN 架構，目的是為了透過文字描述特定物品的特徵，以此產生擬真圖片。鄭佳昇[24] 於 2021 年透過尋找會影響人群分佈的因子，後將各群因子當作 cGAN 輸入預測台北不同區域的人流，並且針對不同區域建立各自的 cGAN 模型。Yi & Babyn[19]於 2018 年利用 GAN 與銳利度檢測網路改善目前處理斷層掃描圖片中光子不足導致雜訊影響診斷的問題。Chen 等人[3]於 2018 年開發新的 Cartoon-GAN 能夠將真實資料的圖片轉換成不同類型的卡通風格圖片，解決以往將圖片轉成卡通風格的滿意度問題。

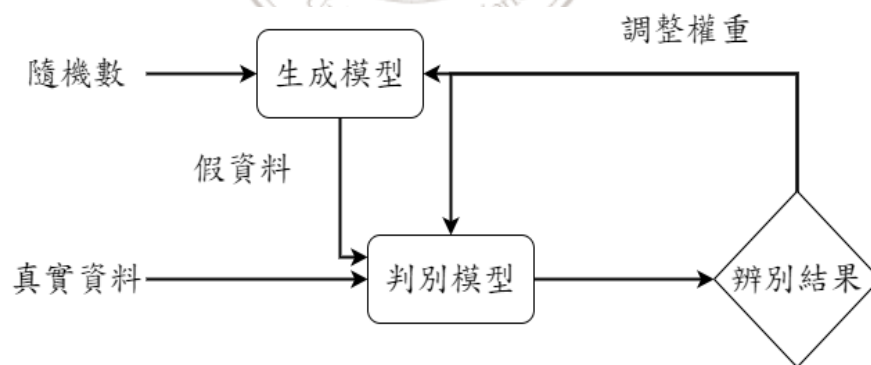


圖 6 GAN 架構圖

第三章 資料集

3.1 遊記資料集

遊記資料集來源於 PTT 日旅版，搜集標題中含有「[遊記]」以及「東京」的遊記如圖 7 所示。時間範圍從 2015-2022 年，透過網路爬蟲所搜集得到的遊記資料集如表 1 所示。資料筆數為 3107 篇。抓取的欄位資料包含文章標題、文章內容。且在疫情爆發以前每一季都至少有 50 篇以上的遊記，如圖 8 所示。

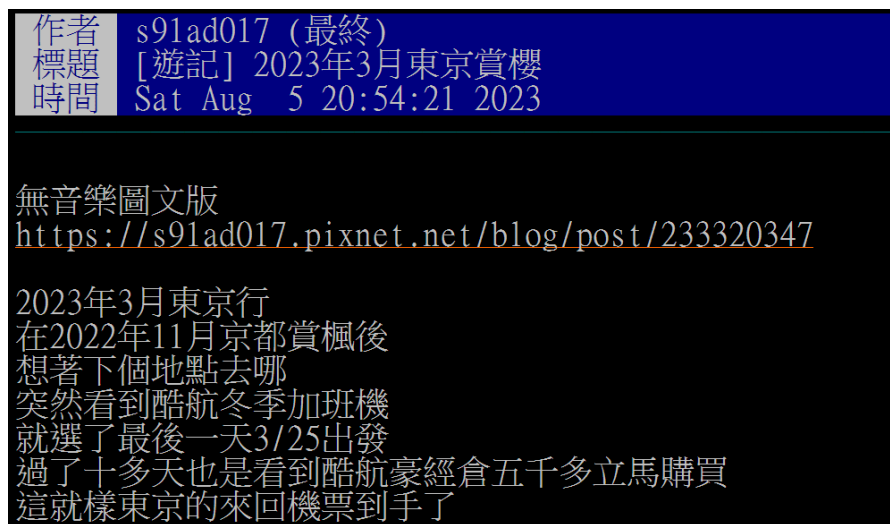


圖 7 PTT 旅遊網誌圖

表 1 遊記資料集

文章標題	文章內容
[遊記]東京 4 天 3 夜遊	待業一段時間，3 月底拿到聘書的隔天，我便立馬買了松山-羽田的機票跟住宿，時間是 4/ 7-14....
[遊記]東京淺草川越台場四天三夜遊	2/12 號 桃園 1 航廈 酷航 GK12 -> 成田第 3 航廈 接駁車到成田第 2 航廈 -> 京成 SKY...
...	...
[遊記]平成進令和出東京咖啡遊	3 月初的某天，看到虎航又在促銷機票，手指頭忍不住的就開始點選便宜的時間，點一點突然發現.....

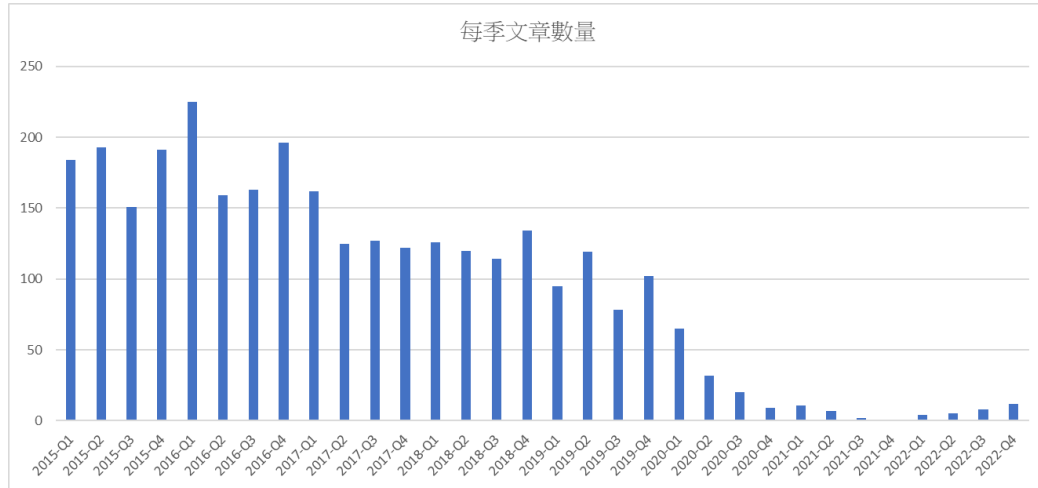


圖 8 每季文章數量直方圖

3.2 景點資料集

景點資料集來源於 Google Map，輸入關鍵字「東京 景點」後，將所有顯示出來的結果全部記錄下來，後利用 Google Map API 提取出景點細節，一併收錄至景點資料集中。景點所在經緯度皆位於東京都(35.313~35.821N, 139.555~139.923E)內，一共有 98 個景點。資料集如表 2 所示。資料欄位包含景點名稱、目前狀態、經度、緯度以及景點類型。

表 2 景點資料集

景點名稱	目前狀態	經度	緯度	類型
東京晴空塔	OPERATIONAL	35.71006	139.8107	['tourist_attraction', 'point_of_interest']
東京鐵塔	OPERATIONAL	35.65858	139.7454	['tourist_attraction', 'point_of_interest']
...
明治神宮	OPERATIONAL	35.6764	139.6993	['tourist_attraction', 'place_of_worship', 'point_of_interest']

欄位「景點名稱」為在 Google Map 上所顯示的正式名稱；欄位「目前狀態」為該景點目前的營業狀態「OPERATIONAL」為目前還在營業，「CLOSED_TEMPORARILY」

為已經停業，空值則為區域劃分的景點，無法衡量營業狀態如：台場、六本木。；欄位「經度」為景點所在地之經度；欄位「緯度」為景點所在地之緯度；欄位「景點類型」則是 Google Map API 上所記載的景點類型標籤。圖 9 為將景點資料集利用地圖視覺化過後的結果，可以看出景點大多數都集中在右側靠近港灣的地方。

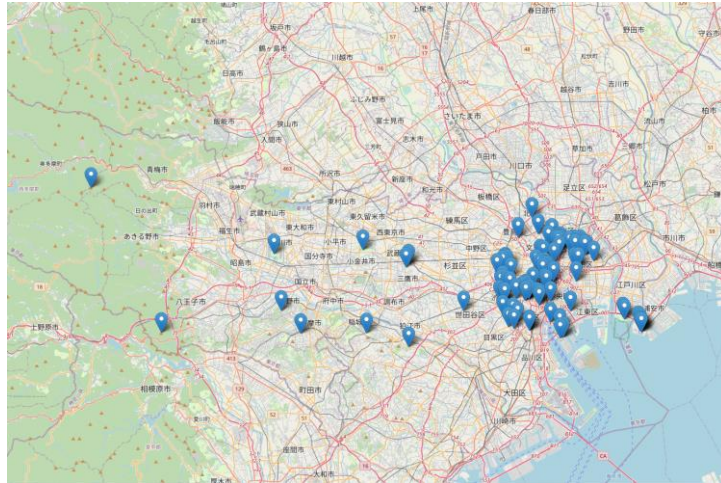


圖 9 景點分佈圖



第四章 研究方法

本研究分為 Offline processing(訓練)與 Online processing(測試)兩個部分，研究方法架構圖如圖 10 所示。在 Offline processing 中可以分成五個步驟。第一步對本研究所搜集的遊記資料集透過 ChatGPT 進行景點順序的提取。第二步將文內出現過的景點標記在網格上，並依照景點數量將時間平均加入到網格中，產生出可以輸入至 GAN 的訓練資料集。第三步將訓練 GAN 產生擬真行程網格。第四步為生成資料評估，目的為評估模型有效性。在 Online processing 中會讓使用者指定當天最想去景點，轉換成網格形式後輸入至訓練好的生成器中，最後我們將輸出結果透過視覺化以及文字的形式返還給使用者。

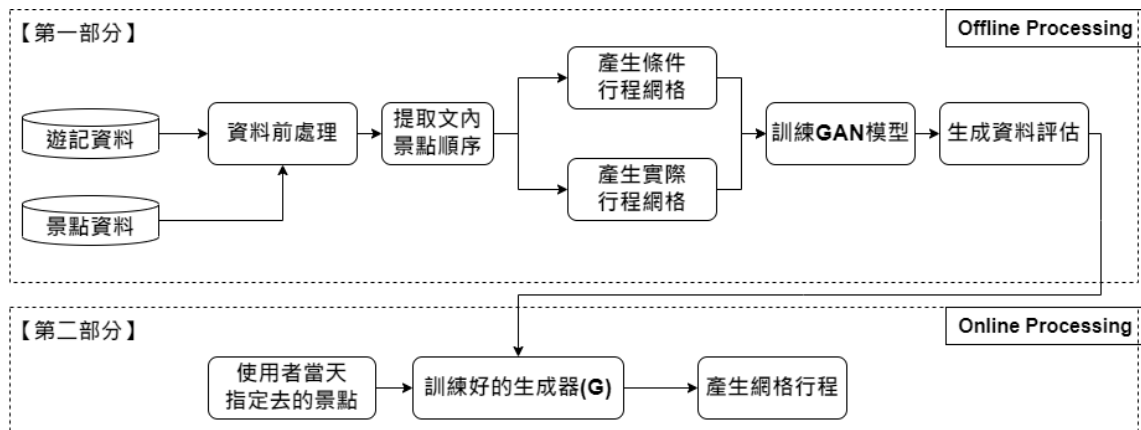


圖 10 研究流程圖

4.1 透過 ChatGPT 進行景點順序提取

4.1.1 文本處理

由於 ChatGPT 在進行計算時會計算 Token 數量，因此本研究需要先将蒐集下來的遊記進行特殊符號以及網址的刪除其結果如表 3 所示，後續才能夠輸入至 ChatGPT 中進行景點順序提取，保留標點符號的原因為經過本研究測試 ChatGPT 需要保留有標點符號能夠判斷出上下文關係才能夠很好的提取景點順序。

4.1.2 ChatGPT 景點順序提取

在使用 ChatGPT 時需要盡量將問題表示明確，本研究在嘗試數次後發現將經過 4.1.1 節後的遊記與「請回傳一個中文 Python 字典,其中值為上述文字中出現的每天景點順序的 List 並用逗號隔開每個景點」一併輸入至 ChatGPT 當中便可獲得可供本研究進行後續處理的景點提取結果，其結果如表 4 所示。

表 3 文章處理前後比較表

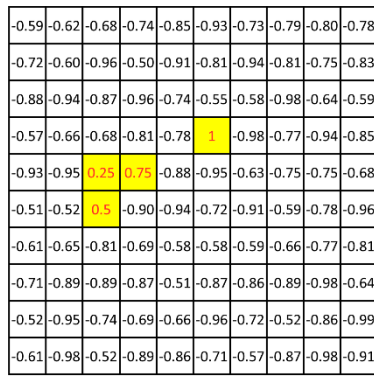
原始文章	處理後文章
從日本回來一個月了.....和大家分享一下 東京+日光 七天六夜，特別自製短片紀念，四分鐘一起遊東京～ https://youtu.be/6f5ScN8q2Yw 對日光可說是念念不忘，儘管不在楓紅最盛時，那種混雜各色的美景，綠黃橘紅，也是一種獨特。.....	從日本回來一個月了.....和大家分享一下，屬於小情侶的自由行。東京日光七天六夜，特別自製短片紀念，四分鐘一起遊東京～對日光可說是念念不忘，儘管不在楓紅最盛時，那種混雜各色的美景，綠黃橘紅，也是一種獨特。.....

表 4 ChatGPT 輸入以及輸出

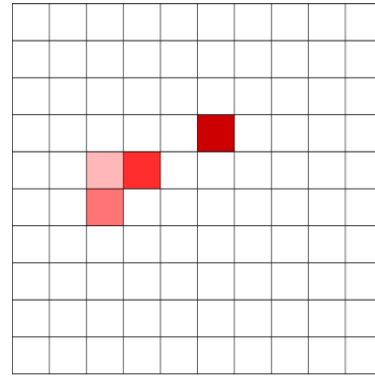
ChatGPT 輸入	景點提取結果
「從日本回來一個月了.....和大家分享一下東京+日光 七天六夜，特別自製短片紀念，四分鐘一起遊東京～ https://youtu.be/6f5ScN8q2Yw 對日光可說是念念不忘，儘管不在楓紅最盛時，那種混雜各色的美景，綠黃橘紅，也是一種獨特。.....」請回傳一個中文 python 字典,其中值為上述文字中出現的每天景點順序的 list 並用逗號隔開每個景點。	{"Day1": ["淺草", "晴空塔"], "Day2": ["明治神宮", "竹下通", "新宿御苑", "秋葉原"], "Day3": ["吉祥寺", "三鷹吉卜力"], "Day4": ["淺草", "日光", "華嚴瀑布"], "Day5": ["湯之湖", "戰場之原"], "Day6": ["東照宮", "二荒山神社", "輪王寺"], "Day7": ["成田", "台北"]} }

4.2 實際行程網格資料

在經過 4.1 節後，本研究依照文內出現的景點名稱與景點資料集做比對，若文內景點有在景點資料集中，則從資料庫取得景點座標軸，若無匹配則將其景點名稱利用網路爬蟲的方式將結果新增至景點資料集當中。在取得景點座標軸後，將某一天出現的全部景點依照其出現的順序轉換實際行程網格資料供 GAN 模型訓練判別器，如圖 11(a)所示，以下將以 10x10 的網格進行示意說明。



(a) 輸入至判別器的實際行程網格



(b)視覺化過後的實際行程網格

圖 11 實際行程網格示意圖

圖 13 為將表 4 當中第三天所出現的景點標記在網格上，顏色深淺則代表遊玩順序，透過轉換網格的方式能夠清楚的知道該天去過哪裡、遊玩順序兩種資訊。然而大多數的遊記並未詳細記錄其遊玩的確切時間，因此本研究透過觀察景點資料集發現東京地區的一般商店、百貨公司以及博物館等營業時間大多數皆從上午十點營業至晚上八點，因此本研究將沒有標註時間的遊記依照其文內出現的順序來平分遊玩時間，其中網格內數值的計算方式如式 1 所示。

$$Ri = Ci * (\frac{1}{n} Ni) \quad (1)$$

不會將時間考慮進去，目的是為了讓生成網路學習到景點遊玩時間以及固定使用者所選定的景點。

表 5 景點組合表

組合
明治神社、新宿御苑
明治神社、秋葉原
明治神宮、竹下通
竹下通、新宿御苑
竹下通、秋葉原
新宿御苑、秋葉原

-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	1	1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

(a)選定明治神宮與新宿御苑為必去景點

(b)選定秋葉原以及竹下通為必去景點

圖 12 景點條件網格示意圖

4.4 建立 GAN 模型

4.4.1 GAN 架構

本研究透過 4.3 節的行程條件網格資料做為 GAN 模型的輸入，利用 GAN 產生出擬真網格資料。生成對抗網路是由兩種網路構成，其分別為生成網路(Generator Network)與判別網路(Discriminator Network)，生成網路以生成假資料為主，其目的是為了讓判別網路無法區分出真資料與假資料，判別網路以判別資料真假為主，其目的是為了識別和者才是生成網路所產生的假資料，在兩種網路不斷的競爭後，最終判別網路會無法判斷生

成網路所產生的擬真資料。本研究的 GAN 模型架構如下圖 13 所示，主要是用來產生符合條件的擬真行程網格。生成網路的輸入為 4.3 節中提到的景點條件網格，最後輸出擬真行程網格，其目標為產生與實際行程網格相似的網格資料，稱為擬真行程網格，如圖 14 所示。

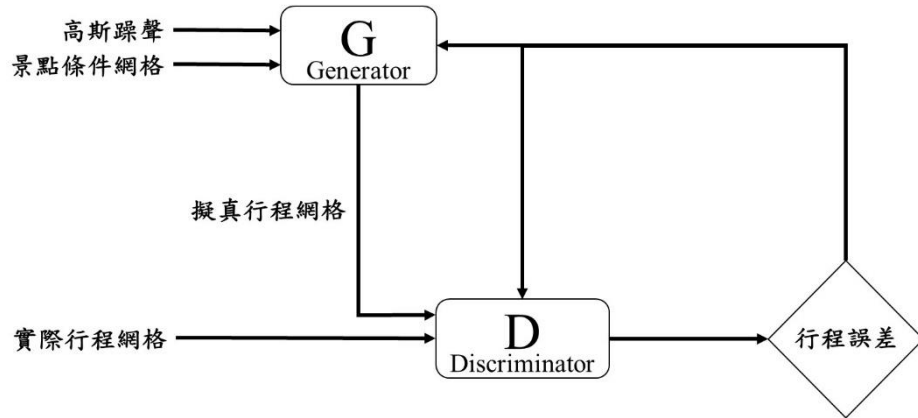


圖 13 本研究 GAN 架構圖

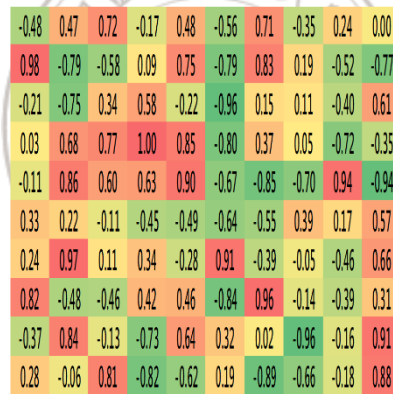


圖 14 擬真行程網格

其中擬真行程網格中的每一個網格數直接由生成器填入，其數值代表的是前往該網格景點遊玩的機率，越高代表前往的可能性越高，在生成器生成擬真行程網格資料過後，將會與真實行程網格一並輸入至判別網路進行訓練，判別網路最終會預測擬真行程網格與真實行程網格之間的差異，並回饋給生成器作為權重修正方向。

下面將介紹本研究所使用的三種不同 GAN 架構，主要是想要透過此種方式確認哪種類型的模型架構更適合本研究目的，而不同的架構所使用的激活函數也將會有所不同，以下介紹時將全部以 10x10 大小的網格進行講解。

4.4.2 全連接層架構

在全連接層架構下，生成器會將高斯噪聲以及景點條件網格一併輸入至攤平層，將兩者合併成一維數據後經過 n 層的全連接層架構如圖 15 所示，可以看到在全連接層的架構底下本研究是先將一維數據縮小後再進行放大，若是在 10×10 的大小底下，則輸出層將會輸出 100×1 的一維向量，之後透過 Reshape 層輸出成擬真行程網格；而判別器的部分就是經過 m 層全連接層，將輸入的行程網格不斷縮小後，最後判斷輸入的行程是真實的還是擬真的；生成器與判別器的激活函數皆是使用 Tanh 函數。

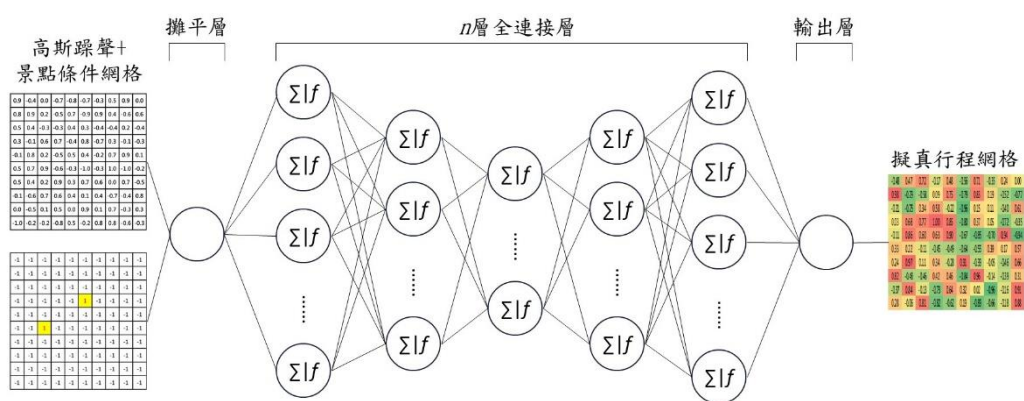
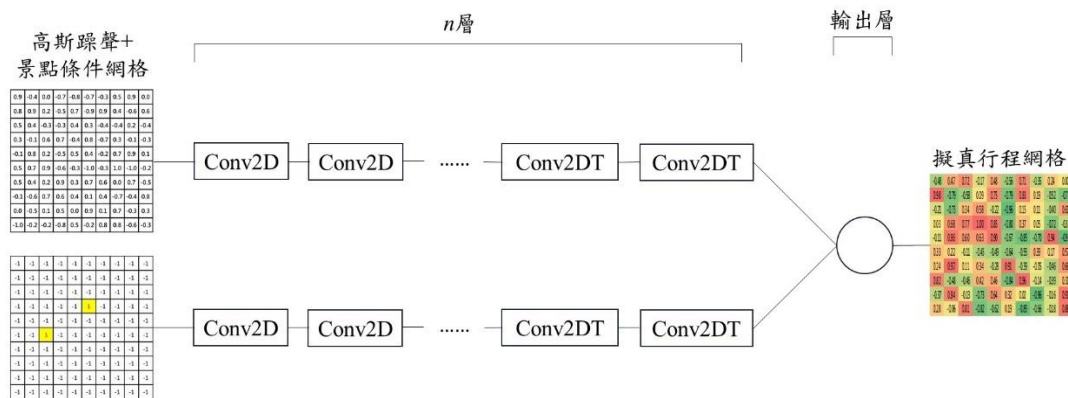


圖 15 生成器全連接層架構

4.4.3 卷積層架構

卷積層架構主要保留了條件景點網格中的空間特徵進行處理，生成器的部分會將噪聲以及景點條件網格分開處理，兩者皆經過 n 層正卷積層(Convolution Layer, Conv2D)，將網格縮小至一定程度後，再經過 m 層反卷積層(Convolution Transpose Layer, Conv2DT)，將網格放大回與輸入相同大小的結果後，兩者再經過計算最後輸入擬真行程網格，如圖 16 所示；而辨別器的部分為經過 t 層正卷積層，將輸入網格縮小後最後判斷輸入行程為真實或擬真；再此架構下所使用的激活函數為 Relu 函數。



4.4.4 殘差網路與最大池化架構

殘差網路架構主要是處理高斯噪聲，使其先經過 n 層的全連接層，最後先輸出一個與輸入大小相同的結果，之後再將景點條件網格輸入至模型與先行處理的結果合併後進行攤平，之後經過一層的全連接層，最後輸出擬真行程網格，如圖 17 所示；而判別器則是經過了 m 層的最大池化層，將輸入的行程進行縮小但其仍留有空間特徵，之後經過 t 層的全連接層，幫助判別器進行判斷輸入行程為真實或是擬真，如圖 18 所示；在殘差網路架構下，兩者的全連接層皆使用 Tanh 激活函數。

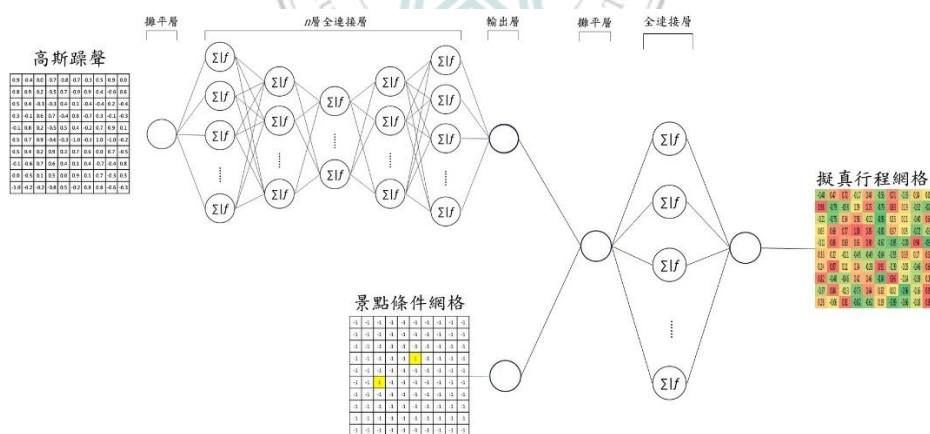


圖 17 生成器殘差網路架構

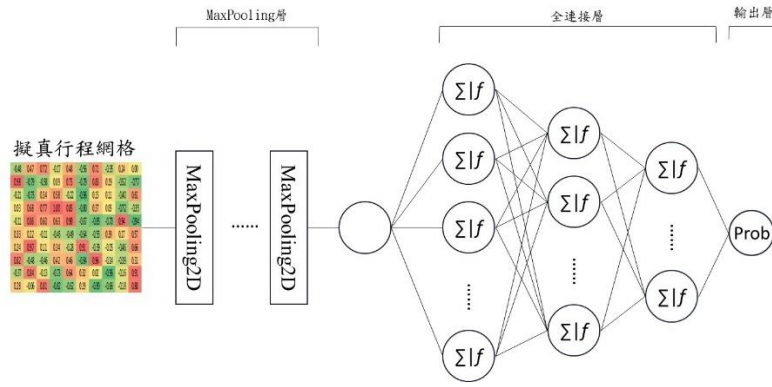


圖 18 判別器最大池化架構

4.4.5 損失函數

本小節將介紹本研究提出的三種 Punish Loss 進行說明，其會針對有實際行程中出現的網格與擬真行程的網格進行誤差計算，讓 Loss 不容易被稀釋，使模型更快進行收斂如式 2；另外還有針對沒有前往的網格的誤差值的平均以及最大值如式 3 與式 4，目的是為了讓模型知道哪一些網格是不會前往的，進而讓生成結果更貼近實際狀況。

$$TE = \frac{1}{N} \sum_{i=1}^N |R_i - F_i| \quad (2)$$

$$error1 = \frac{1}{n} \sum_{j=1}^n |F_j + 1| \quad (3)$$

$$error2 = \max(|F_j + 1|) \quad (4)$$

公式 2 的 i 代表真實行程網格中數值大於 0 的第 i 個網格， R 代表真實行程網格， F 代表擬真行程，兩者相減過後取絕對值，並將每個網格進行平均，得到代表行程網格誤差的 TE ，其值越接近 0，代表生成網路所產生的行程網格資料越符合實際行程網格，最後回饋給生成網路以及判別網路進行更新，只針對實際行程網格當中有前往的網格進行計算的原因為，若計算所有網格的誤差將會出現有前往的網格誤差雖然很大，但卻因為其他不重要的網格誤差很小而被稀釋掉。

4.5 模型評估

本研究的目的是在於利用生成對抗網路產生擬真的行程資料，因此將訓練過後的生成對抗網路使用測試資料集將產生的擬真行程網格與實際行程網格透過 MAE (Mean Absolute Error)進行績效評估，其中 MAE 如公式 5 所示。若是 MAE 指標較小，則將會使用針對特定區域計算的改良 MAE 指標，如公式 6 所示，其中 i 為實際有前往的網格及其附近八格之所有網格編號，其示意如圖 19 所示，其中 i 表示圖 19 當中黃色以及橘色部分， j 表示圖 19 白色部分，但本研究會在計算白色部分時乘上兩倍的懲罰權重，以此來表示有多少在黃橘色範圍外的網格是有數值的，此做法是因為橘色部分在本研究算可以被接受的範圍，因此當傳統 MAE 指標較小時，才會針對較為細部的網格進行計算，以便得知真實情況；此外，本研究會將實際行程網格與擬真行程網格當中網格數值小於 0 的值改為 0 以方便計算。

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (5)$$

$$newMAE = \frac{1}{N} \sum_{i=1}^N |R_i - F_i| + \frac{1}{n} \sum_{j=1}^n |R_j - F_j| * 2 \quad (6)$$

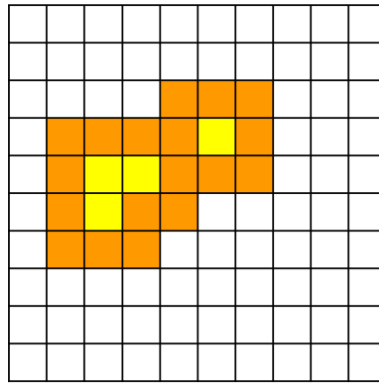


圖 19 改良 MAE 計算示意圖

第五章 實驗模擬

此章節將介紹本研究的驗證實驗，包含 5.1 節清洗過程、5.2 節建立網格、5.3 節不同架構的 GAN 對預測結果的影響、5.4 節績效探討。本研究所有實驗都是在 64 位元作業系統上運行 Windows 10 的 Python 和 3.6GHz 的 Intel Core i7-9700 處理器，48GB 記憶體以及 NIVDIA GeForce RTX 2080 Ti 上處理的。

5.1 清洗過程

本節將會分成 (1) 遊記文章清洗、(2) ChatGPT 景點順序提取、(3) 景點爬蟲以及(4) 篩選區域過程三個小節進行介紹

5.1.1 遊記文章清洗

本節將說明本研究在清洗資料的過程中發現或刪除的文章，再觀察資料集的過程中發現若文章標題帶有「推薦」二字的文章內容大多數都沒有遊玩順序，如表 6 所示。總計有 42 篇文章標題含有推薦字眼，其中 40 篇都是在推薦店家或伴手禮，故需刪除。而由於東京迪士尼可以從早上玩到晚上，不符合本研究所需，因此透過觀察發現有 150 篇文章標題含有「東京迪士尼」、「迪士尼」的文章，其中有 120 篇內容皆是在討論如何更有效的遊玩東京迪士尼，如表 7 所示，故需刪除，之後刪除 12 篇的空值欄位。在清洗完上述所提到的文章過後剩餘 2935 篇遊記，由於本研究需要透過 ChatGPT 來蒐集遊記當中遊客每一天的行程，因此遊記當中需要存在天數詞，如第一天、第二天、Day1、Day2 等，因此會先行判斷文章內是否含有天數詞，判斷如表 8 所示。接著本研究發現有 2127 篇不帶有任何日期相關詞語的遊記，不符合本研究所需，故刪除。

表 6 文章標題含有「推薦」的文章內容

文章標題	文章內容
[遊記] 東京成田機場伴手禮推薦	Royce 是北海道札幌的巧克力品牌，台灣也有賣 不過貴很多日本的零食辦手禮真的超多，如果在 市區看到這些零食．．．

表 7 文章標題包含「迪士尼」的文章內容

文章標題	文章內容
[遊記] 東京迪士尼海洋玩樂攻略 玩樂表演一次包	這次東京自助之旅的重點，準備進攻從小到大一直想去的～東京迪士尼海洋樂園一下舞濱站開始即感受迪士尼的氣氛東京迪士尼園區非常廣大．．．

表 8 文章天數詞提出

文章標題	文章內容	是否含有天數詞
[遊記] 2015 東京！ (D2 築地、淺草和服)	2/7(六)，day2：築地(八千代早餐)、淺草寺(逛街、和服體驗)、上野(買藥妝)．．．	是
[遊記] 利久牛舌 東京 車站 丸大樓冰雪奇緣 聖誕燈	久聞利久牛舌的盛名但我們來了日本那麼多次卻沒吃過這實在太說不過去了所以晚上特地坐車到了新宿品嚐	否

5.1.2 ChatGPT 景點順序提取

本研究所使用的 GPT 模型為 Gpt-3.5-turbo，其單次提問與回答總共最多可以接受 4096 個 Token，以平均來看，1 個 Token 約可以寫 0.5 個中文字或 1 個英文字，但因為每個中文字所消耗的 Token 數量不一樣，因此需要先行進行計算，避免輸入產生錯誤或 ChatGPT 只回答到一半 Token 數就用完了的情況，計算 Token 數量後的結果示意如表 9 所示，可以看到文章長度與 Token 數量並不一致，然而上述提到提問與回答總共最多能有 4096 個 Token，因此必須篩選 Token 數量至一個區間，使其回答不會中斷，由於提問時還需要加上本研究所設計的問題：「請回傳一個中文 Python 字典,其中值為上述文字中出現的每天景點順序的 List 並用逗號隔開每個景點」，因此本研究經過測試後，將數量定在介於 500 至 3700 之間，經過此操作後剩餘 599 篇遊記，之後將剩餘的 599 篇遊記透過 ChatGPT 進行景點順序提取，其提取結果如表 10 所示，可以看到，在提取結果中，ChatGPT 能夠將一些店鋪的名字提取出來，也能夠依據上下文自行判斷其前往的順序，將景點順序提取出後就要與景點資料及進行比對，並使用爬蟲來尋找查找景點經緯度；本研究考慮到成本問題使用 gpt-3.5-turbo，未來亦可使用更高級的模型進行提取。

表 9 文章 Token 計算

文章標題	文章內容	文章長度	Token 數量
[遊記] 2016 48 小時 東京追櫻去	搶到威航紅眼，只有 48 小時也是要衝一下第一次櫻花季來，把握時間努力踩點 4/5Day1 橫濱大岡 . . .	203	134

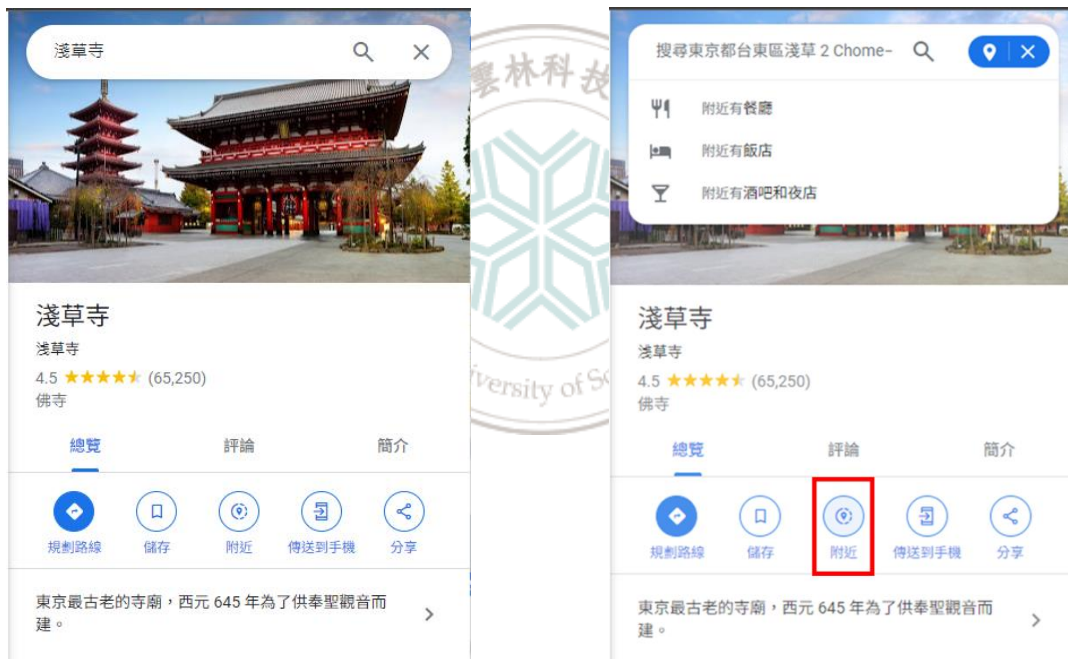
表 10 ChatGPT 完整輸入及輸出表

文章內容	提取結果
<p>「主要想分享吃的食物及簡單心得！6/2 四搭香草 am7:25 飛機住池袋的東急 stay 首發吃了 RedRock 高田馬場店因為是下午 4 點左右到所以沒有排就順利進去了 lucky 辦啾井（因為肉跟辦啾頭髮一樣高）牛排井辦啾井的肉勝，牛排肉比較韌但這家店是這次旅行的前幾名！下次來還想吃之後前往吉祥寺，吉祥寺如同版上的大家所說，真的滿好逛的！我好喜歡雜貨和小物～但是男友覺得還好 QQ 晚餐前看到 misterdonut 忍不住進去，點了以下三種不過都還好 XD 除了巧克力那圈還不錯，另兩個都有的偏甜晚餐いせや總本店本店去吃的時候已經幾乎什麼都沒了只有兩三種便宜的串燒可點（一個是蔥，另兩種都是雞肉類，不過一串 80yen 真的好便宜啊！）所以有有點雞腿和燒賣～燒賣大顆但普串燒便宜！cp 值不錯雞腿很大一個拿起來啃滿爽的但是這家店吃完沒什麼記憶點下次應該不會來吃了 XD 第一天結束～～」請回傳一個中文 python 字典,其中值為上述文字中出現的每天景點順序的 list 並用逗號隔開每個景點</p>	<p>{Day1: ['RedRock 高田馬場店', '吉祥寺', 'misterdonut', 'いせや總本店本店']}</p>
<p>「從日本回來一個月了.....和大家分享一下，屬於小情侶的自由行。東京日光七天六夜，特別自製短片紀念，四分鐘一起遊東京～對日光可說是念念不忘，儘管不在楓紅最盛時，那種混雜各色的美景，綠黃橘紅，也是一種獨特。每一幕都值得回憶，湯之湖到戰場之原的愜意漫步，半月山的陡峭難行.....很可怕不要問！反正我人生不會再爬第二次半月山了抖關於器材這是我們第一次帶運動攝影機出門，男友還改裝了手機專用的 SmoothQ，我真的很佩服他！請看和 Kevin 老師的合照，脖子那台 XD 關於後製從兩百多段毛片中東挑西「剪」同樣是我的第一次，不專業但已盡力，算是對這趟旅程有個交待了！行程如下 Day1:台北>成田>飯店>淺草>晴空塔 Day2:明治神宮>自由之丘>中目黑 Day3:吉祥</p>	<p>{ "Day1": ["淺草", "晴空塔"], "Day2": ["明治神宮", "自由之丘", "中目黑"], "Day3": ["吉祥寺", "三鷹吉卜力"], "Day4": ["淺草", "日光", "華嚴瀑布"], "Day5": ["湯之湖", "戰場之原"], "Day6": ["東照宮", "二荒山神社", "輪王寺"], "Day7": ["成田", "台北"] }</p>

寺>三鷹吉卜力 Day4:淺草>日光>華嚴瀑布 Day5:湯之湖
>戰場之原 Day6:東照宮>二荒山神社>輪王寺 Day7:成田
>台北」請回傳一個中文 python 字典,其中值為上述文字
中出現的每天景點順序的 list 並用逗號隔開每個景點

5.1.3 景點爬蟲

本研究將 5.1.2 節提取結果欄位中出現過的景點使用 Google Map 爬蟲，其爬蟲的過程如圖 20 所示，**首先將每一天視作獨立的單位**，假設第一天第一個景點為淺草寺，則在搜尋欄上輸入「淺草寺」，待其出現結果後，蒐集淺草寺的經緯度後，按下「附近」按鈕，接著在搜尋下一個景點，直至第一天所有景點被查找完；透過 Google Map 的「附近」按鈕即可使該景點做為起點而查詢下一個景點。若是碰見有複數個搜尋結果，則默認點擊結果當中的第一個結果。



(a)使用搜尋欄搜尋淺草寺

(b)點擊「附近」搜尋下一個景點

圖 20 爬蟲步驟示意圖

5.1.4 篩選區域過程

經過爬蟲過後本研究將所有蒐集到的景點視覺化至地圖上如圖 21 所示，可以看到還是有位於東京都以外的景點，這是因為有些遊記有某幾天在東京有某幾天在東京都外，也會出現當天從東京都出發前往其他縣市或從其他縣市回東京都的行程，本研究依據當天景點的位置分為三類：(1) 東京都內旅遊、(2) 跨縣市旅遊以及 (3) 東京都外旅遊，其統計過後的長條圖如圖 22 所示，可以看到東京都旅遊的占比較多，其次是跨縣市旅遊再來是東京都外旅遊，因此本研究保留東京都內旅遊的行程做為後續研究的資料集；而後續統計了景點數量並繪製成盒鬚圖如圖 23，本研究保留景點數量位於 2 到 5 之間的行程，經過篩選之後，所有出現在該種類行程的景點視覺化如圖 24 所示，可以看到大多數的景點都集中再東京 23 區內，而為避免網格數量過於龐大，本研究將範圍縮小至如圖 25 所示，最後設定網格邊長為 1 公里乘 1 公里，得到一個 14x16 大小的網格，後續將利用此大小來建立網格。

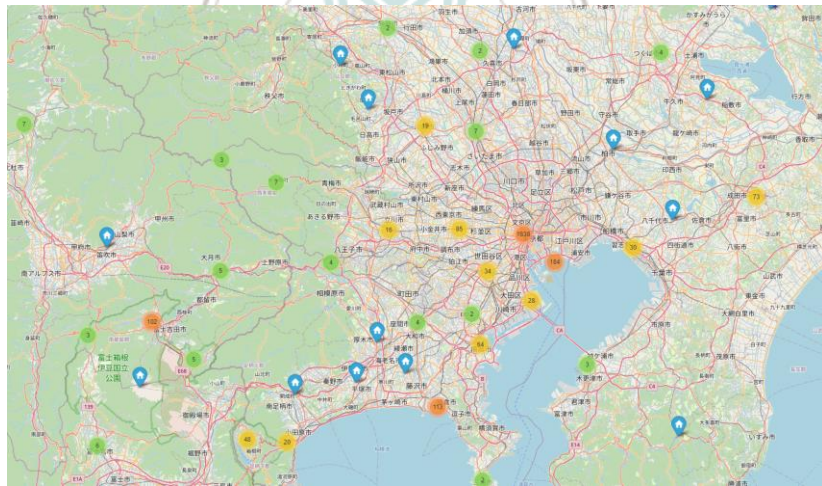


圖 21 遊記中出現過所有景點分布圖

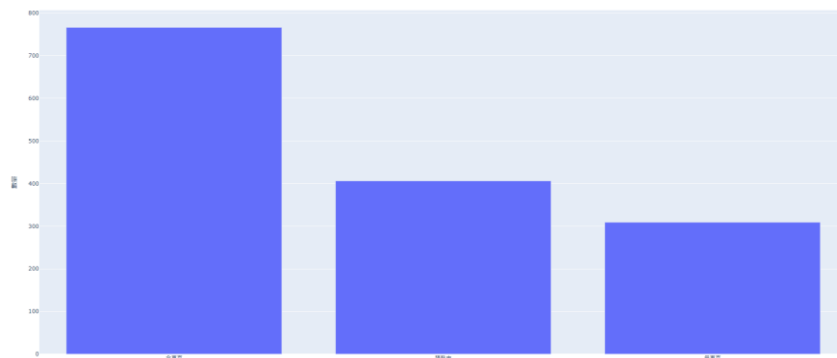


圖 22 旅遊型態長條圖

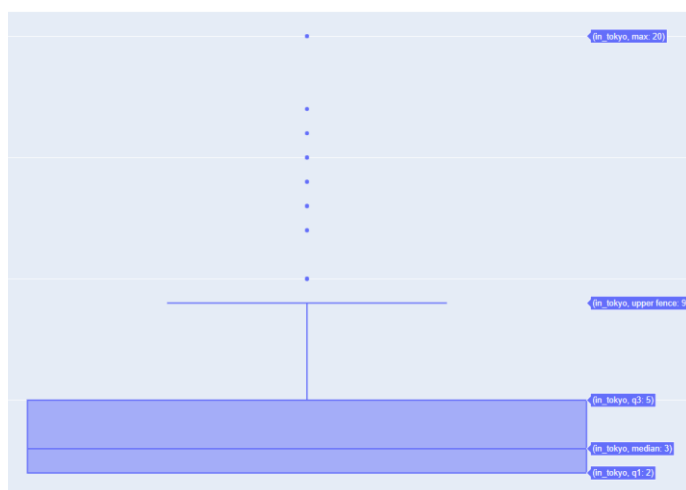


圖 23 行程景點數量盒鬚圖

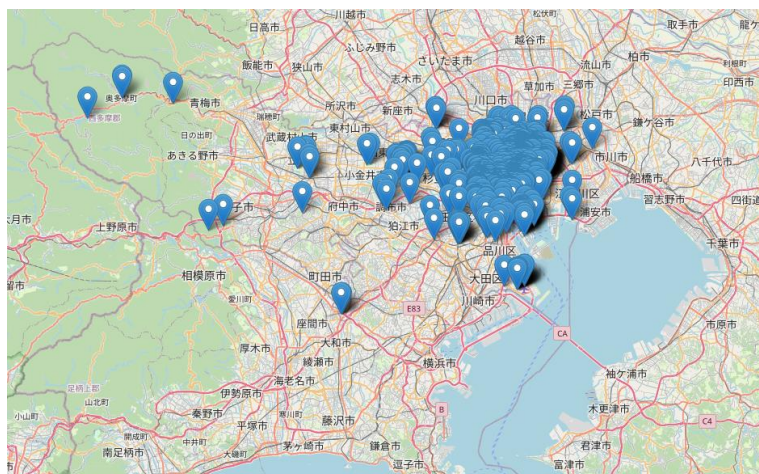
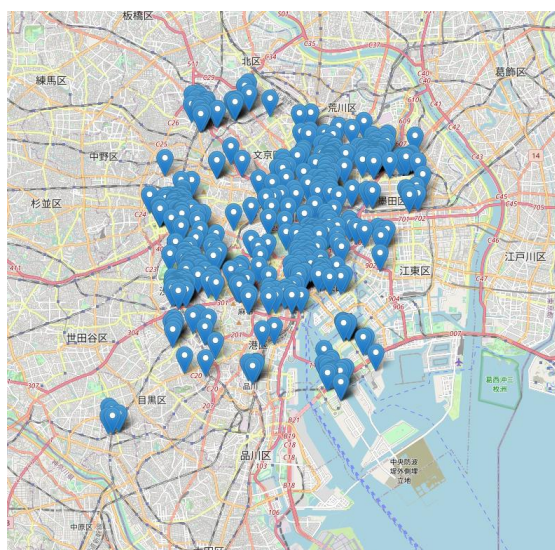
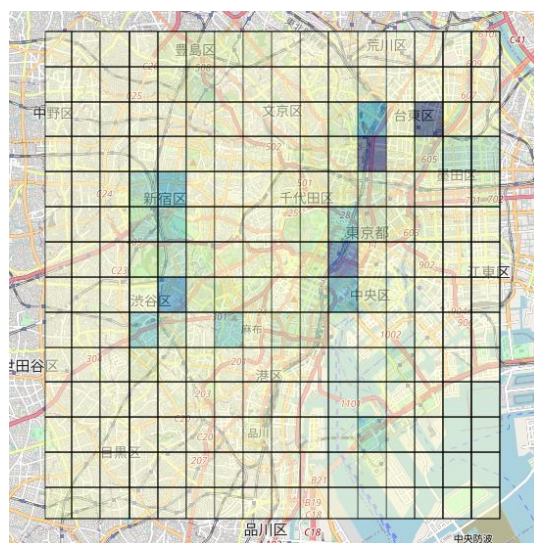


圖 24 東京都內旅遊景點分布圖



(a) 景點分布圖



(b) 景點熱區圖

圖 25 東京 23 區景點分布圖

5.2 建立網格

本研究依據 5.1.4 小節篩選的區域過後獲得了一個 14x16 的網格地圖，本研究使用此範圍來進行景點條件網格以及實際行程網格的建立，其中過程如 4.2 與 4.3 節介紹的一樣，將當天行程景點進行兩兩一組的組合後，將其位置標記為一其餘為-1，即可獲得景點條件網格；將景點數量進行比例平分後，依照順序加總至網格地圖上，並將沒有前往的景點網格填入-1 至-0.5 之間的浮點數，即可獲得實際行程網格。

5.3 不同架構的 GAN 對預測結果之影響

本節介紹本研究嘗試不同架構下的 GAN 對預測結果不同的影響，其中分為 5.3.1 全連接架構、5.3.2 卷積網路架構以及 5.3.3 殘差網路架構，5.3.1 至 5.3.3 的模型參數為 Batch 為 128、Epoch 為 7000、優化器使用 Adam 其 Learning-rate 為 0.0002。

5.3.1 全連接架構

此種架構底下的生成網路以及判別網路皆使用全連階層做為模型架構，生成網路的全連接層神經元數目為 256、128、64、32、16、32、64、128，輸出層為 224 之後轉成 14x16 大小，判別網路的神經元數目為 512、256、128、32、8，最後輸出判斷與是否為真實行程；圖 26 與圖 27 分別為此架構訓練至第 6000 Epoch 與 7000 Epoch 的結果，圖 28 為 Loss 曲線，可以看到 Loss 曲線在大約 1000 Epoch 後就趨近於平緩，這代表判別網路在此種架構的模型以及現有的損失函數下，無法有效的判斷出真實的行程以及由生成網路生成的擬真行程，經過訓練過後生成器依舊會輸出此種類型的圖片代表，待表判別器認為此種類型的行程網格與實際行程網格，以圖 26 與圖 27 來說，可以看到兩者之間並沒有太大的差別，代表生成器只需要生成這種類型的行程即可騙過判別器；對此本研究增加兩種 Punish Loss 藉此改善模型結果，其兩種損失函數如公式 3 與公式 4 所示，本研究將 4.4.2 節的損失函數與新的兩種損失函數各別乘上 0.33 的權重後，再次進行訓練，其第 7000 Epoch 結果與 Loss 曲線為圖 29 與圖 30，可以看到在經過兩種新的損失函數後，對模型學習的幫助提升非常大，此階段的模型已經能夠順利生成非常相似於真實行程的擬真行程了。

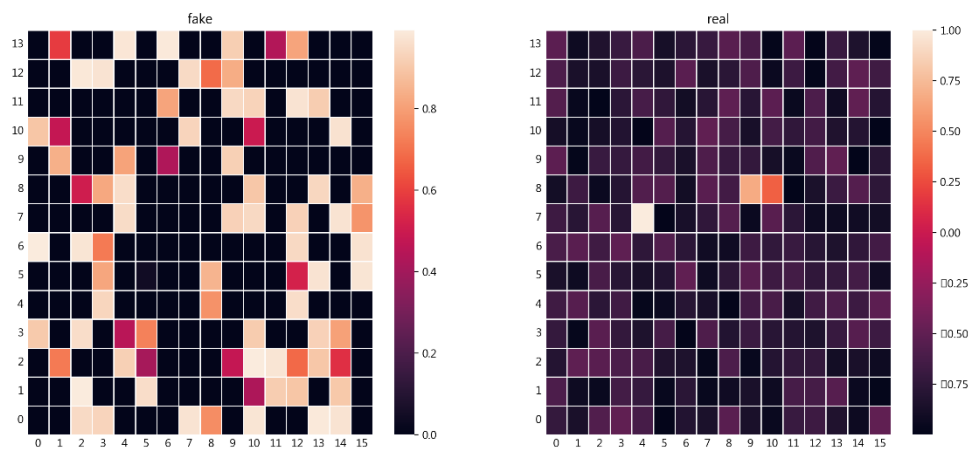


圖 26 單一損失函數全連接層第 6000 Epoch 訓練結果

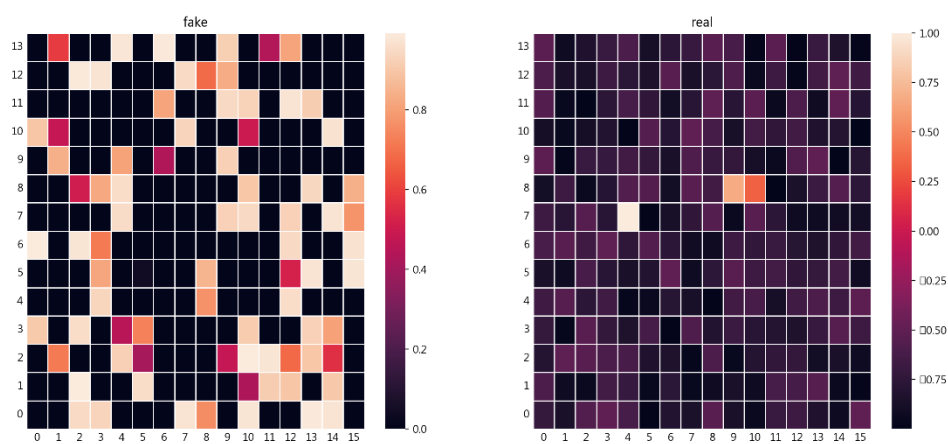


圖 27 單一損失函數全連接層第 7000 Epoch 訓練結果

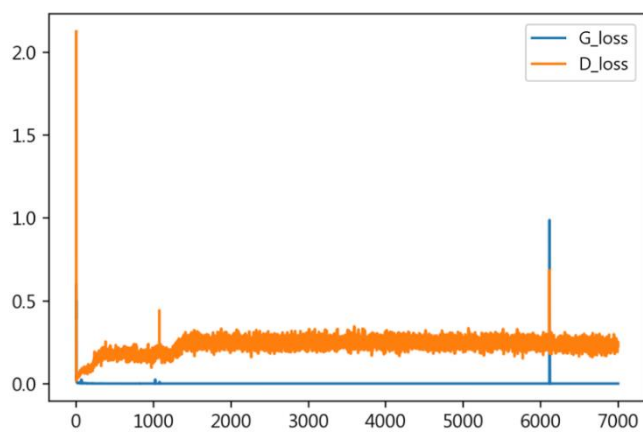


圖 28 單一損失函數全連接網路 Loss 曲線

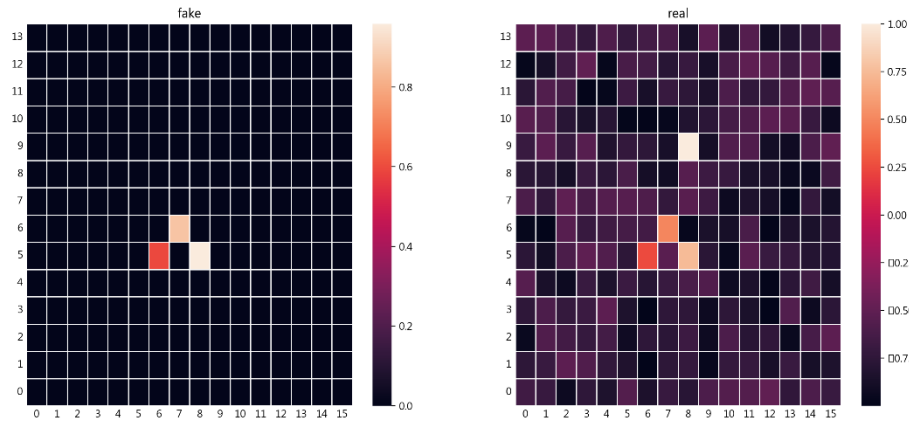


圖 29 三種損失函數全連接層第 7000 Epoch 訓練結果

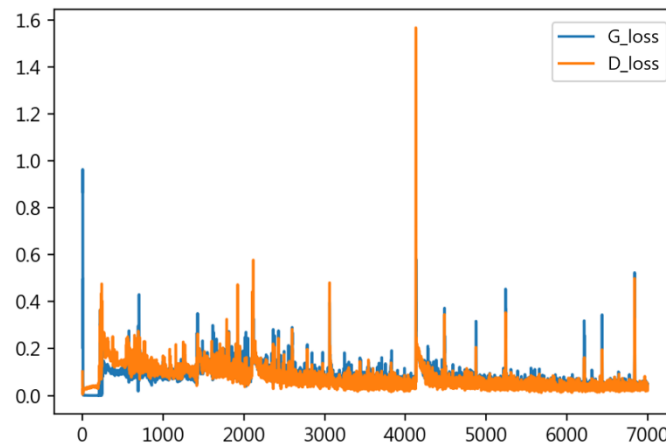


圖 30 三種損失函數全連接層 Loss 曲線

5.3.2 卷積網路架構

卷積網路架構的生成器是先經過四層的卷積層將輸入縮小後，再透過四層反卷積層與一層反卷積輸出層將輸出放大至原有大小；而判別器的話是經過六層的卷積層將輸入縮小後，最後輸出判斷是否為真實行程。卷積網路架構在 6000 Epoch 與 7000 Epoch 的結果為圖 31 與圖 32，Loss 曲線為圖 33 所示，可以看到雖然相較於全連接層來說生成器已經可以生成沒有過多白點，然而透過圖 31 與圖 32 兩張圖可以看到，生成器生成的行程在右下角出現的躁點，這可能是鑑於卷積網路架構會將特徵放大縮小，導致判別器將右下角的區塊視作行程的一部分，讓生成器認為一個擬真的行程就應該包含右下角，然而如果拋開右下角不談，還是能夠看到擬真與實際的某些網格是有對上的，對於 Loss 曲線圖來說在大約 1000 Epoch 之後兩條曲線產生了分歧，這代表兩者並沒有互相影響，

導致生成器在訓練後期只會產生一樣的結果來欺騙判別器，這很好的解釋了為何生成器在生成的時候都會加上右下角的區塊來欺騙判別器；而增加 Punish Loss 的 6000 Epoch 與 7000 Epoch 為圖 34 與圖 35，其結果可以看到相較於只使用的單一種的損失函數，使用了三種的損失函數並沒有改善模型，反而讓生成器只會生成單一結果，這代表此種結果即可騙過判別器，學習不佳；使用三種損失函數的 Loss 曲線如圖 36 所示，大概在 2000 Epoch 之前 Loss 兩條曲線皆呈現下降趨勢。

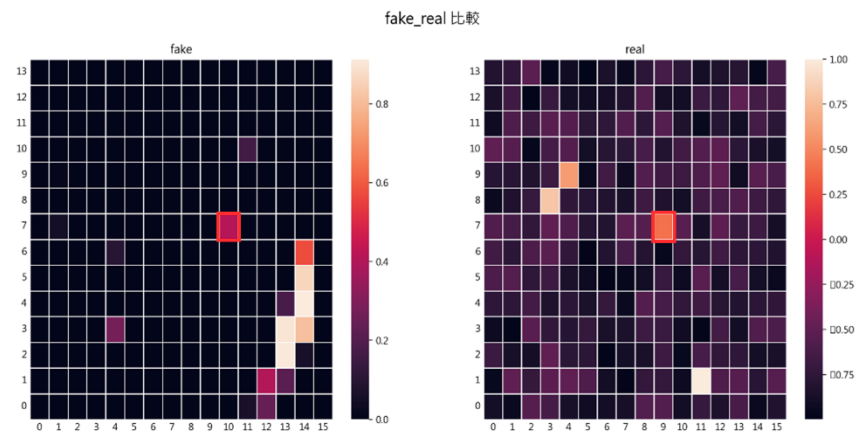


圖 31 單一損失函數卷積網路第 6000 Epoch 訓練結果

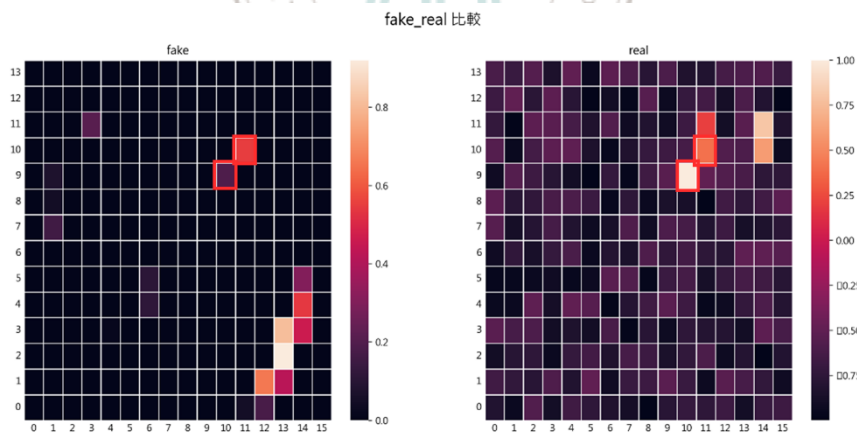


圖 32 單一損失函數卷積網路第 7000 Epoch 訓練結果

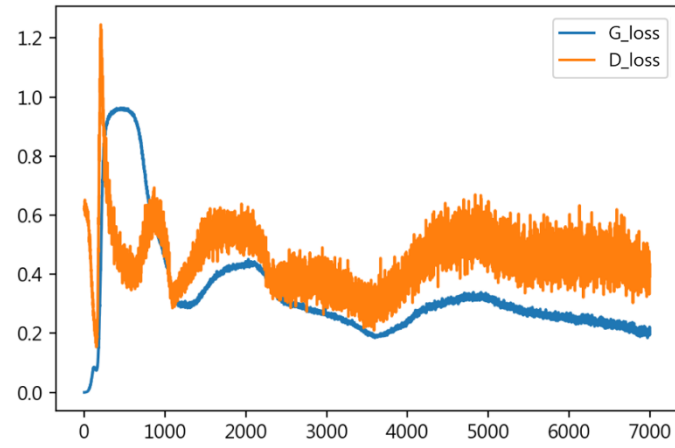


圖 33 單一損失函數卷積網路 Loss 曲線

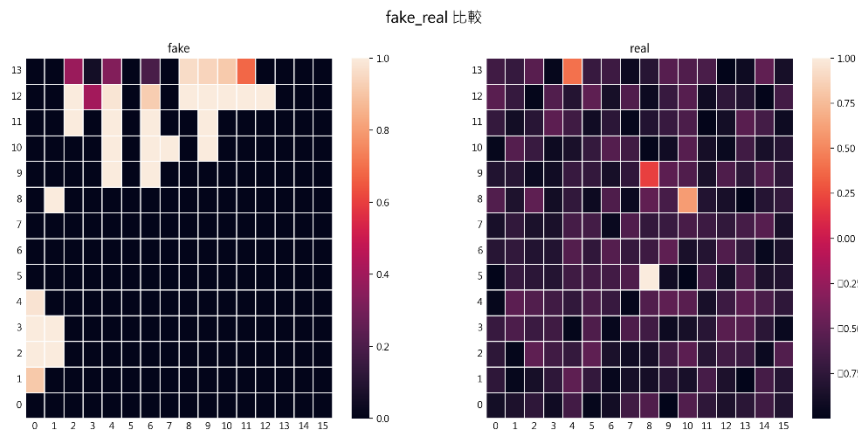


圖 34 三種損失函數卷積網路第 6000 Epoch 訓練結果

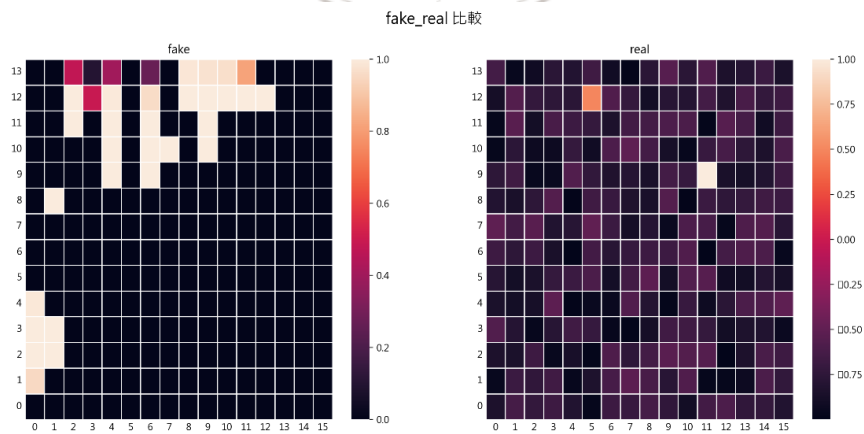


圖 35 三種損失函數卷積網路第 7000 Epoch 訓練結果

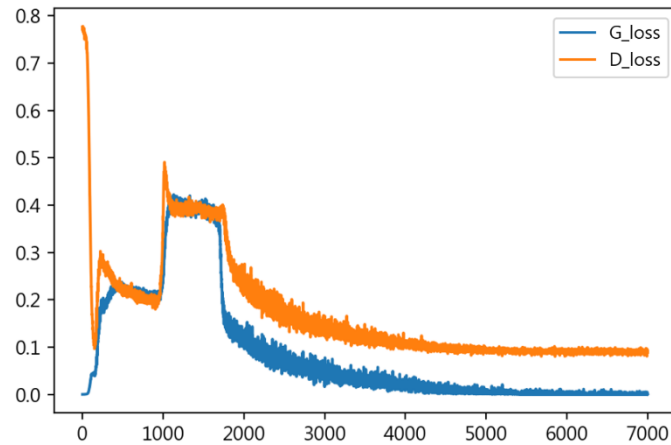


圖 36 三種損失函數卷積網路 Loss 曲線

5.3.3 殘差網路與最大池化架構

本研究最後嘗試對生成器建立殘差網路架構，其內容為先讓二維噪聲經過神經元數目為 256、128、64、32、16、32、64、128 的全連接層架構，之後與景點條件網格進行合併後攤平，輸出層為 224 的全連接層，其目的是讓景點條件網格再最後才影響生成器的輸出；而判別器網路架構為先經過 4 層的最大池化(Max Pooling)層，使得輸入縮小為 10x12 但依然保持地圖特徵後進行攤平，最後經過神經元數目為 32、8 的全連接層最後輸出判斷是否為真實行程，使用最大池化層的原因為由於前往的景點網格對於整個空間來說可能就集中在某些區域的其中某幾個網格，剩餘的網格可能會干擾模型判斷，因此使用最大池化層將網格縮小，但保持原有的特徵狀態讓模型進行參考，其在第 7000 Epoch 時的訓練結果為圖 37 與圖 38，分別為較好與較壞的結果，圖 39 為 Loss 曲線。

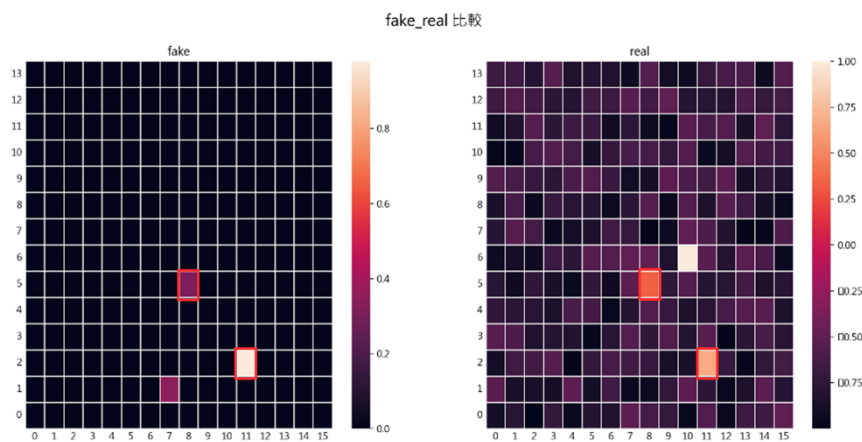


圖 37 殘差網路與最大池化第 6000 Epoch 訓練結果

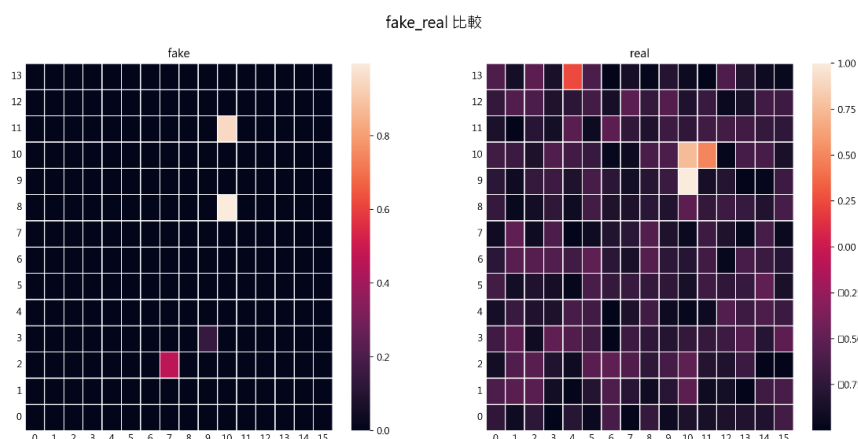


圖 38 殘差網路與最大池化第 7000 Epoch 訓練結果

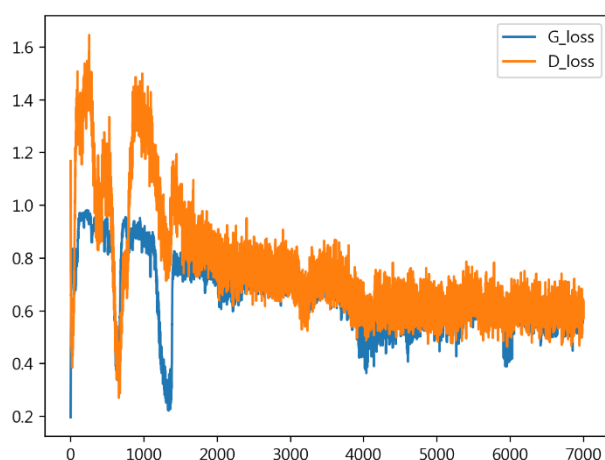


圖 39 殘差網路與最大池化 Loss 曲線

在此架構底下由於在判別器部分有將特徵部位進行不影響相關位置的放大後，模型整體的學習效率增高，大約在 2000 Epoch 兩者互相開始影響，值至 7000 Epoch 兩者之間的 Loss 值約在 0.5 左右，這對生成對抗網路來說是很好的一種表現，兩者互相影像，最終趨於平緩。

5.4 績效探討

本小節針對 5.3.1 至 5.3.3 節之績效探討，使用的是在訓練資料外的測試資料集，其績效指標使用 4.5 節所提到的公式 5 來當作衡量指標，若當中有特別低的績效，則會另外使用公式 6 來進行探討，績效表為表 11，在使用測試資料對五種類型的架構進行測試過後發現，原本結果看起來較好的殘差網路架構在測試階段的績效相較於卷積網路來的較差，其本研究認為原因很可能是由於先針對躁聲進行全連接層訓練，導致在測試的

時候躁聲輸入一但有偏差就很容易產生錯誤的結果，其中殘差網路於測試資料集上的結果為圖 40 與圖 41 所示；最後本研究將會針對使用三種損失函數的全連接層與單一損失函數的卷積架構進行進一步的績效探討，其結果如表 12 所示；從傳統的 MAE 績效來看，使用三種損失函數的全連接層與單一損失函數的卷積網路並沒有太大的差異，但經過改良的 MAE 過後兩者個差異能夠很明顯地分別出來。

表 11 五種類型輸入準確率比較表

	全連接架構+ 單一損失函數	全連接架構+ 三種損失函數	卷積架構+ 單一損失函數	卷積架構+ 三種損失函數	殘差網路
MAE 績效	15.769194	4.549623	5.510591	7.077367	12.792118

表 12 改良 MAE 績效比較表

	全連接架構+ 三種損失函數	卷積架構+ 單一損失函數
newMAE 績效	22.757072	47.281432

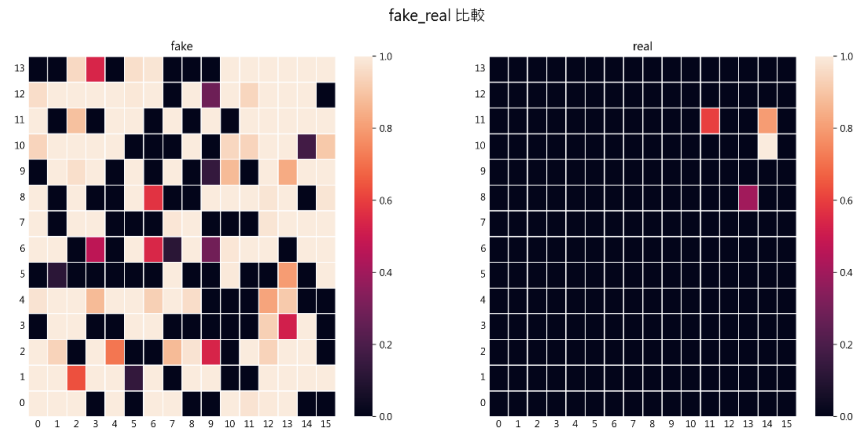


圖 40 殘差網路與最大池化測試資料預測結果

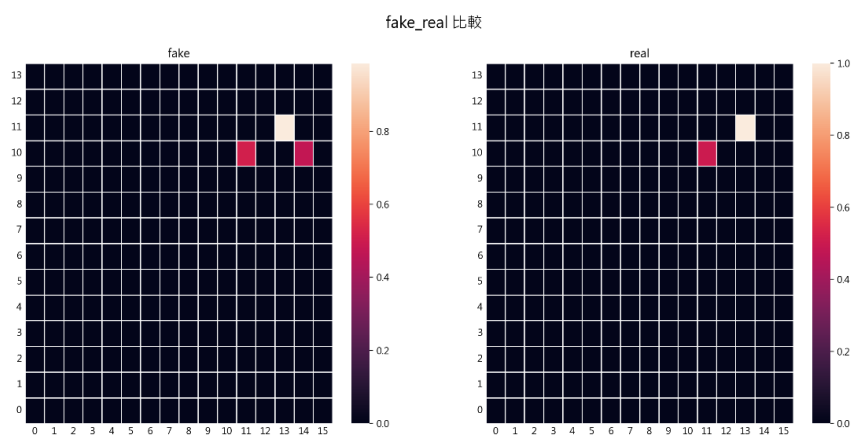


圖 41 殘差網路與最大池化測試資料預測結果



第六章 結論與未來研究

為解決過往旅遊推薦系統問題，本研究針對過往研究中的不足之處進行改善，提出一個新的框架用於實現旅遊推薦系統。預計使用過往旅客分享的遊記資料作為目標資料集，景點之詳細內容為輔助資料集，使用 GAN 進行旅遊行程的推薦，利用 GAN 的隨機，就能夠提供旅客在選擇某些景點的情況下，安排出符合所選條件的不同行程，本研究目前嘗試三種不同的網路架構以及三種不同的損失函數進行預測比較，發現針對全連接層架構，使用三種不同的損失函數能夠非常有效的提升模型產生擬真行程的能力，而在卷積網路架構由於可能會受到其他周圍景點網格的影響，因此生成結果都會帶有雜訊出現；最後是殘差網路架構，在此架構底下本研究發現在訓練的過程中，此種類型的網路架構能夠有效的進行擬真行程網格的生成，但在測試階段時，由於躁聲與景點條件網格的不同，導致模型生成結果好壞參半，未來可以再針對這部分進行探討。

目前本研究所設計的模擬資料缺少了針對旅行上非常重要的因素—跨區旅遊，即目前模擬資料只能夠推薦單一區域的行程，然而日本交通網路發達，交通時間大幅度的縮短後，使得跨區域的旅遊行程得以實現，未來若能開發出不同架構，讓空間計算從大到小，就能夠實現跨區域的旅遊景點與產生更多樣化的行程以供遊客參考。

在未來本研究可以分成幾部分進行精進，包含了(1)在資料集的方面可以尋找更明確，更精簡的網站進行爬蟲，即可以大幅度的省略資料前處理的步驟。(2)在演算法的部分由於東京地區太過龐大，可能可以藉由網格分群加速運算。

參考文獻

- [1] Aydın, Ö., & Karaarslan, E. (2022). OpenAI ChatGPT generated literature review: Digital twin in healthcare. Available at SSRN 4308687.
- [2] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., & Askell, A. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33, 1877-1901.
- [3] Chen, Y., Lai, Y.-K., & Liu, Y.-J. (2018). Cartoongan: Generative adversarial networks for photo cartoonization. *Proceedings of the IEEE conference on computer vision and pattern recognition*,
- [4] Coelho, J., Nitu, P., & Madiraju, P. (2018, July). A personalized travel recommendation system using social media analysis. In *2018 IEEE International Congress on Big Data (BigData Congress)* (pp. 260-263). IEEE.
- [5] Fenza, G., Fischetti, E., Furno, D., & Loia, V. (2011). A hybrid context aware system for tourist guidance based on collaborative filtering. *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*,
- [6] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2020). Generative adversarial networks. *Advances in neural information processing systems*, 27,
- [7] Goyal, T., Li, J. J., & Durrett, G. (2022). News summarization and evaluation in the era of gpt-3. *arXiv preprint arXiv:2209.12356*.
- [8] He, Z., Wu, Z., Zhou, B., Xu, L., & Zhang, W. (2015). Tourist Routs Recommendation Based on Latent Dirichlet Allocation Model. *2015 12th Web Information System and Application Conference (WISA)*,
- [9] Huang, Y., & Bian, L. (2009). A Bayesian network and analytic hierarchy process based personalized recommendations for tourist attractions over the Internet. *Expert Systems*

with Applications, 36(1), 933-943.

- [10] Ishanka, U. P., & Yukawa, T. (2018). User emotion and personality in context-aware travel destination recommendation. 2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA),
- [11] Jeblick, K., Schachtner, B., Dexl, J., Mittermeier, A., Stüber, A. T., Topalis, J., Weber, T., Wesp, P., Sabel, B., & Ricke, J. (2022). Chatgpt makes medicine easy to swallow: An exploratory case study on simplified radiology reports. arXiv preprint arXiv:2212.14882.
- [12] Jia, Z., Yang, Y., Gao, W., & Chen, X. (2015). User-based collaborative filtering for tourist attraction recommendations. 2015 IEEE international conference on computational intelligence & communication technology,
- [13] Lu, E. H.-C., Chen, C.-Y., & Tseng, V. S. (2012). Personalized trip recommendation with multiple constraints by mining user check-in behaviors. Proceedings of the 20th International Conference on Advances in Geographic Information Systems,
- [14] Nitu, P., Coelho, J., & Madiraju, P. (2021). Improvising personalized travel recommendation system with recency effects. *Big Data Mining and Analytics*, 4(3), 139-154.
- [15] Phillips, T., Saleh, A., Glazewski, K. D., Hmelo-Silver, C. E., Mott, B., & Lester, J. C. (2022). Exploring the use of GPT-3 as a tool for evaluating text-based collaborative discourse. Companion Proceedings of the 12th, 54.
- [16] Qin, C., Zhang, A., Zhang, Z., Chen, J., Yasunaga, M., & Yang, D. (2023). Is ChatGPT a general-purpose natural language processing task solver? arXiv preprint arXiv:2302.06476.
- [17] Reed, S. E., Akata, Z., Mohan, S., Tenka, S., Schiele, B., & Lee, H. (2016). Learning what and where to draw. *Advances in neural information processing systems*, 29.
- [18] Subramaniaswamy, V., Vijayakumar, V., Logesh, R., & Indragandhi, V. (2015). Intelligent travel recommendation system by mining attributes from community

- contributed photos. *Procedia Computer Science*, 50, 447-455.
- [19] Yi, X., & Babyn, P. (2018). Sharpness-aware low-dose CT denoising using conditional generative adversarial network. *Journal of digital imaging*, 31(5), 655-669.
- [20] Yu, Y., Wang, H., Sun, S., & Gao, Y. (2017). Exploiting location significance and user authority for point-of-interest recommendation. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*,
- [21] 中華民國交通部觀光局. (民國 108 年). *108 年臺灣觀光年報*.
https://admin.taiwan.net.tw/upload/contentFile/auser/b/annual_2019_hm/01_3_Domestic-Travel.html
- [22] 日本國土交通省觀光廳. (令和元年). *宿泊旅行統計調查*.
<https://www.mlit.go.jp/statistics/file000008.html>
- [23] 台灣趨勢調查. (民國 108 年). *國外旅遊行為調查*. <https://www.twtrend.com/trend-detail/travel-2019/>
- [24] 鄭佳昇. (民國 110 年). *基於 3D-RCL 與條件式生成對抗網路產生未來時刻人群分布*. 國立雲林科技大學.