Utilizing NLP and Hybrid Recommender Approach to Enhancing Personalized Travel Planning

MSc Research Project
MSc in Artificial Intelligence for Business

Shih-Ting Huang
Student ID: x23266287

School of Computing
National College of Ireland

Supervisor: Victor del Rosal

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Shih-Ting, Huang
Student ID:	X23266287
Programme:	MSc in Artificial Intelligence for Business **Year:** 2025
Module:	MSc (Research) Practicum/Internship Part 2
Supervisor:	Victor del Rosal
Submission Due Date:	
Project Title:	Utilizing NLP and Hybrid Recommender Approach to Enhancing Personalized Travel Planning
Word Count:	6349 **Page Count** 20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……………………………………………

Date: ……………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Utilizing NLP and Hybrid Recommender Approach to Enhancing Personalized Travel Planning

Shih-Ting, Huang
X23266287

Abstract

Nowadays, people in increasing numbers prefer independent and personalize trip rather than book an itinerary from travel agencies or follow the guidebooks. However, the process of search and collect travel information could be time-consuming and lacking efficiency sometimes. Although the progressing of technology such as ChatGPT can help to mitigate this situation due to the understanding of human language, it may lack real-time data and misunderstanding in some conditions. Therefore, this project aims to create a simple travel chatbot that combines multiple APIs in order to collect real-time data with ChatGPT to realizing natural language and associating users to organize their travel itineraries. In the end, we create a synthetic dataset with 15 variables to evaluate the travel chatbot system as well as utilizing a correlation matrix for analyzing the relationship between each other in order to understanding if the system could fit travelers' demands and build a better travel chatbot in the further experiment.

Keywords: Travel, Recommendation System, Artificial Intelligence, Large Language Models

1 Introduction

In recent years, the number of tourists has been increasing, according to UN Tourism (2025), it would be continuously growth in the future. With the ever progressing of technology and the popularity of the internet, the way people planning trips and searching for information are more convenient. For instance, Google search, some travel planning website and large language models (LLMs) such as ChatGPT are all make trip planning easier than usual. Compared with the past that people relied on travel agencies, guidebooks, and personal recommendations for trip planning, the majority of information can be found through internet nowadays. Traditionally, traveller needs to book trips from travel agency because they can provide organized itinerary including flight ticket, hotel, transport, food and tourist spots. However, based on research from Taiwan in 2024, the mode of traveling has been changed, 75% of respondents choose independent travel rather than book an organized itinerary, especially when travel to the U.S., Japan, Korea, Oceania and Thailand. A report form Forbes (2024) shows that the trend for solo trip has increased and be predicted to keep growing. Solo trip has changed from a very niche type of activity to a popular way (totallydublin, 2025). Another trend for traveling is personalized and customaries trip, in the world with interconnected and continues advancing technology, people tend to seek more unique and special experience for traveling (Pruvo, 2025). Apparently, those reports prove that the travel mode has changed from traditional way which depends on travel agency to solo trip, independent trip and personalized trip.

Planning an individual trip requires searching for various information, including flight, hotel, transportation and attractions. This process could be time consuming and tiring sometimes; thus, it would be more efficient when using technology to deal with these problems. Noticeably, LLMs of Artificial Intelligence (AI) have been advancing and wildly use in recent years. This technology could improve trip planning as well as to provide more complete and customaries information such as weather conditions, transportation, and pricing. Forbes (2022) mentioned that AI can find the best flights and hotels for the most competitive price and provide customers personalized experience for travelling. In 2025, a report shows that AI is revolutionizing the global tourism industry, which provides opportunities for serving personalize and increasing customer experience (Shreeansh et al., 2025). Besides from trip scheduling, AI can, as well, associating with language and cultural obstacles (Nagar et al., 2024), which could be very useful for travelers to obtain information without language barriers. It is inconvenient and time-consuming that people have to install various apps and visit plenty of websites for their travel information; therefore, a recommender chatbot which can allow user using their natural language to communicate and obtain the information they require for the trip is quite convenient and useful (Alotaibi et al., 2020). With the help of AI, travelers around the world can understand the travel information even in different language, they can easily access data about their destination by reading location articles and checking hotel prices without boundaries (Thanh et al., 2023).

Natural Language Processing (NLP) is one of the AI related subfields, which can understand and response in human way by using machine learning and deep learning technology, providing users better experience when interact with the system. Implement NLP can help businesses understand customer preferences, market conditions and public opinion easier, better and faster (Cole Stryker and Jim Holdsworth, 2024). In 2022, one research used NLP to analyze customer reviews for enhancing prediction on tourist preference. The result showed that whether the review was positive or negative, the prediction had high accuracy (Meira et al., 2022). Byron (2023) mentioned that NLP is a dominate technology for improving recommender system in order to enhance the performance of the suggested output. Utilizing NLP technology to analyze contents from social media and reviews to improve content filtering, enhancing personalize and contextually aware recommendations (Roobini et al., 2024). Binabdullah and Tongtep (2021) mentioned that searching for travel information from experts, books, websites is quite tedious; therefore, utilizing NLP in recommender system can highly increase the efficiency for choosing suitable tourism activities.

Recommender System is one of an AI related usage as well that provide products, services or suggestions to people by using big data analytics and machine learning (ML) algorithms. The utilization of this technology can help user noticed the products they might not notice before. Moreover, the recommender system can provide personalized user experiences. There are three different types of recommendation algorithm, including content-based filtering, collaborative filtering and hybrid recommendation approach (Rina and Cole, 2024). All of them would be describe in the following paragraph.

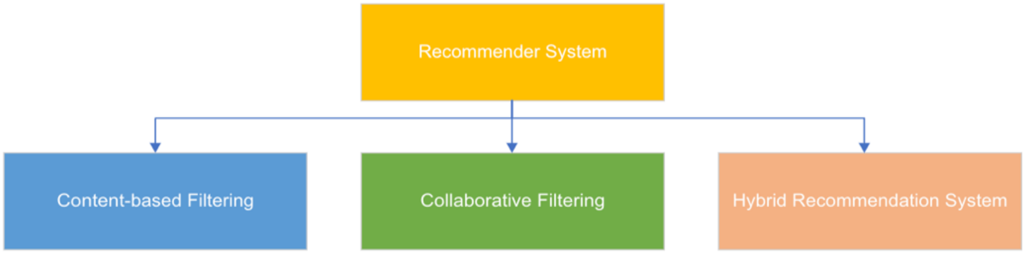

Figure 1: Recommender System

Content-based Filtering

A content-based filtering provides recommendations based on item attributes. This algorithm assumes that if users like A product, it might have chance they would like other products have similar features as product A. The attribute of item includes color, price, category as well as some keywords and tags assigned by metadata. Content-based filtering can avoid cold start problem when having new users due to the item-oriented characteristic. However, it could have problem with new item, because it might lack of connection with original products (Rina and Cole, 2024).

Collaborative Filtering

Collaborative filtering provides suggestions based on the preferences of user group. Therefore, this algorithm is easier having cold start problem to new user. There are two different methods of collaborative filtering, which are memory-based and model-based.

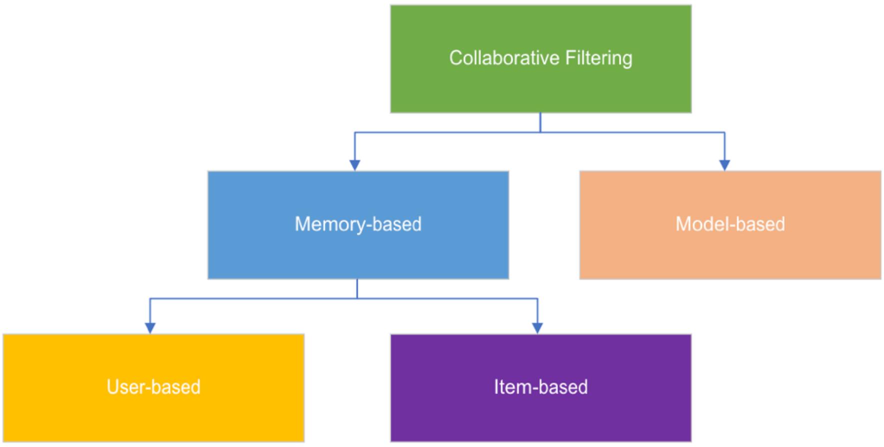

Figure 2: Collaborative Filtering Recommender System

Memory-based

Memory-based includes user-based and item-based which are both related to historical data. The former one calculates the similarity between users and then provides recommendations to each other. For instance, when A user bought product A, B user also bought product A, the system would give user A the other products that user B like. As for item-based, which is analysis the similarity between items, and then recommend those similarity products to users. For example, user A bought product A, the system would provide other products have high similarity with product A to the user.

Model-based

Model-based algorithm provides recommendations based on the model that create from historical data. And then utilize the model to do the suggestion. This algorithm could provide user items that they have not found before.

Hybrid Recommendation System

Hybrid recommendation system merges collaborative filtering and content-based filtering, which can enhance the efficient of the recommender system. This is also the method we are going to implement in the project.

Despite these technological advancements, current systems are lack of comprehended recommendation for all the needed information related to travelling. This research aims to explore how to combine recommender system and travel related data to enhance personalized travel planning and optimize itinerary recommendations. In the end we will compare the traditional search and LLM to realize the benefits and drawbacks between them to improve the recommendation system and user satisfaction.

2 Related Work

This paragraph would describe some related previous works, compare the advantages and disadvantages as well as the justification for these papers.

With the continuous advancement of NLP, recommendation system in increasing numbers have been integrating NLP and machine learning technology to increase user experience and accuracy. Akhilesh et al. (2023) combine multiple NLP methods in the product recommendation system, including Bag of Words, TF-IDF, Word2Vec, and IDF-Weighted Word2Vec. Moreover, they also implement image features to enhance content-based recommendations. Similarly, Dixit et al. (2023) developed a movie recommendation chatbot that utilizes content-based filtering, a “soup of words” approach, and cosine similarity to suggest films based on user preferences such as genres, actors, and directors. Unlike traditional recommenders, this chatbot moves forward to recognize mood or variation in preferences, rather than being limited to a specific type of content.

However, these systems have some limitations, The system by Akhilesh et al. lacks real-time data which could be crucial for dynamic e-commerce environments where user preferences can change rapidly. While the chatbot by Dixit et al. provides dynamic, emotion-aware recommendations, it does not consider users’ past preferences that might lead to less personalization for returning users. This paper builds a dataset for storage movie data, which could be a drawback and bottleneck for dataset limitation. Alotaibi et al. (2020) developed a travel chatbot called “Smart Guidance” for Jeddah, Saudi Arabia, using NLP and machine learning techniques. The chatbot combine the external API to provide real-time data, such as weather information from Weatherstack and location details from other sources. This improves the limitation of the traditional chatbots that lack latest information, ensuring the provided information are up-to-data. However, currently it only supports Arabic language, which would be the limitation for non-Arabic-speaking tourists.

To improve traditional recommender system, Komal Londhe et al. (2024) developed a system called "Itinerary Generator", which combines conversational AI and web scraping to generate travel suggestions including destinations, activities, hotels, and flights based on users’ preferences, budget, and duration. This system can help users to improve their planning process making it more efficient and personalized; however, scalability issues and potential for overwhelming recommendations could be drawbacks of it. Similarly, Lin et al. (2022) improved Collaborative Filtering Algorithm (CFA)-based systems to reduce information overload and enhance recommendation accuracy. By using fusion strategy that combine the average rating from users and the minimum rating from the item to calculate a balance result from overall satisfaction and individual dissatisfaction. However, this paper used a dataset from single city which might have limitation when implement to other cities.

Nan et al. (2022) aimed to improve the accuracy and efficiency of travel recommendations while reducing data processing overheads. To ensure the accuracy and satisfaction of itinerary, customers' travel history and their preferences have been considered. This study improves the accuracy of recommendation by using data mining and collaborative filtering as well as user preferences and travel histories. However, the dependence on historical data could elicit the cold start problem and then struggle with provide accuracy suggestions. Furthermore, integrating multiple advanced techniques increases system complexity and computational demand. Cheng (2021) also utilized historical travel data to recommend points of interest (POIs) by analysing users' interest themes and distance constraints. The model utilizes directed weighted graph to represent tourist spots and their connections, helping users organize their interest themes and the distance more accuracy and efficient, increasing the user satisfaction. The lack of consider other factors such as traffic conditions, weather, or user fatigue could be the disadvantage of this algorithm and the reliance on historical data introduces the same cold start limitation.

To solve the problem of outdated information in traditional recommender systems, Linus (2018) applied data mining, user modelling, and constraint-based recommendation algorithms. This system considered social media and other digital footprints to provide highly personalized travel recommendations and allow more accurate and dynamic user modelling. While this enhances data freshness and user relevance, the reliance on users' historical travel information could elicit cold start problem as well as the raise of privacy concerns for data using from social media and other digital footprints. Nethmin et al. (2023) introduced "TourBuddy," a system for Sri Lanka tourism that integrates collaborative filtering, random forest, CNNs, and image processing to recommend destinations, food, activities, transport as well as cultural elements for a holistic experience. However, this system only focuses on Sri Lanka may have some limitations of its applicability in other countries or cities. Vysotsky et al. (2019) developed "Mandry," which aggregates travel data using Firebase Realtime Database, providing personalized and real-time recommendations. The offline service allows users check the information in anytime they needed; however, the reliance on external data sources may be the limitation if the sources are not regularly update or maintained

LLM, such as ChatGPT, has been significantly improved the process of travel planning by obtain personalized recommendations, streamline itinerary creation, and provide real-time updates. Nikas et al. (2025) designed a system combining traveller profile surveys, LLM-generated recommendations, and satisfaction evaluation to improve travel planning. Harmanpreet et al. (2024) built a recommendation system that combine and compare personal model, including user profile, preferences (e.g., hobbies, cuisine), and personal concepts (e.g., pet name) and generic model that without personal information to investigate the effectiveness of the LLMs as a personal agent, showing that the suggestion with personal judgements is better than generic data. Zhang et al. (2025) integrated LLMs with ACT-R-inspired cognitive modeling to simulate human-like planning based on preferences and contextual awareness. This system provides personalized recommendations, human-like planning, high flexibility and scalability, but demands high computing resources and lacks real-time/localized data. Yang et al. (2025) created a tool-augmented agent that generates multi-city plans, evaluated through automatic and human metrics. Although it can plan trip with multiple days and cities as well as fit user goals and generated plans (e.g., budget, pace, interest), it lacks real-time data integration and may hallucination.

To better align recommendations with user behavior, Tianming et al. (2025) proposed a persona-based embedding framework that adapts to user travel styles such as cultural explorer or budget traveller. The system can combine with any other LLMs without retraining the model even though it could have cold start problem when lack of historical data and noisy preference data. Song et al. (2024) introduced a retrieval-augmented generation (RAG) framework, TravelRAG that mitigate hallucinations and improves grounding by extracting structured facts from multiple sources, this research use NER (Named Entity Recognition). The system enhances semantic understanding but increases processing costs due to its complex architecture. Y. Chen et al. (2025) developed a system that can provide justifies and explains the reason for suggesting the attraction of recommendations by using trajectory-aware prompting and supports multi-turn dialogue. It improves engagement and relevance but still faces hallucination and cold start issues.

To enhance personalized travel recommendations, Aribas and Daglarli (2024) integrated LLMs like ChatGPT with personality models including openness, conscientiousness, extraversion, agreeableness and neuroticism in a hybrid recommendation system. Deep personalization and natural language output are the merits of this system, but it requires private data and risks hallucination. Afridi (2025) developed a serendipity-focused recommendation system combining LLMs and smart transportation, revealing their potential through a user study. Banerjee (2025) creates a recommendation system by using LLMs and RAG to enhance Tourism Recommender Systems (TRS) for sustainable city trips. Turki Alenezi (2022) proposed the NATP model using topic-model-based representations of travel personalities and destinations, improving recommendation ranking quality and gain a better consequence.

Zhiyu Zhang (2024) developed an AI-based tourism recommendation system in terms of recommendation accuracy, user satisfaction, and system performance to evaluate the real-world performance by using AI-based recommendation algorithms, machine learning frameworks (TensorFlow/PyTorch) and real-time data processing (Apache Kafka, Apache Flink). While it provides high recommendation accuracy (85.3%) and precision (82.7%) as well as high user satisfaction (average 4.7/5), the reliance on large volumes of historical user data could having cold start problem and potential data privacy issue. Similarly, Badouch & Boutaounte (2023) also face cold start problem and data sparsity in the personalized travel recommendation systems. However, it provides tailored recommendations to individual preferences and real-time data by using dynamic adaptation.

Fararni et al. (2021) implemented hybrid filtering, big data, and AI to build a recommender system. This system also integrated real-time data and semantic web technologies. With the help of hybrid algorithm, this system mitigates cold start problem; however, it might increase development effort due to the integration of multiple algorithms and data pipelines. Aggrawal et al. (2025) integrated AI, machine learning algorithms and NLP to create a recommendation system to provide better personalize suggestions. It combines dynamic adaptability to adjust itineraries for weather, traffic, or closures as well as improving user satisfaction by using sentiment analysis to prioritize highly rated POIs. However, the reliance on user data from social media could have privacy problem and cold start problem as well.

3 Research Methodology

In this project, we create a simple program that combine Google API and ChatGPT to collect attraction data as well as weather API, flight API to collect weather and flight information. In the end we organize all the collected data by ChatGPT to generate an itinerary of the trip that traveler want. Furthermore, we compare the traditional Google search and ChatGPT for planning itinerary by analyzing synthetic data. The result of this comparison can used to be the reference of future work to improve the program. The framework of this project, the parameter of synthetic data and the collected data from other APIs would be provided below.

The flow chart below shows the whole process of the travel itinerary planning system, which starts with user input in human language, and then attraction data collected from Google and ChatGPT, weather data from open weather and flight data from Amadeus. In the end we used ChatGPT to combine all the collected data to generate an itinerary that fit user's demand.

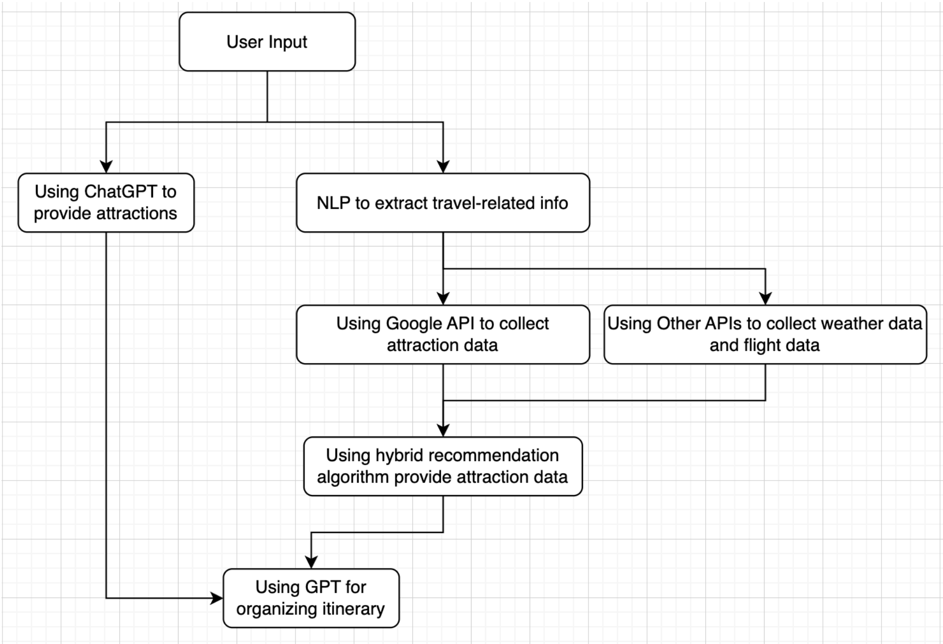

Figure 3: Flow chart of the project

The following tables are the collected Data from APIs (Table 1) and the parameters for the comparison between Google and ChatGPT (Table 2). The former includes attraction data, traveler review from Google, weather data from open weather and flight data from Amadeus. The latter has 15 variables, including user_id, age, tech_savviness, travel_frequency, query_complexity, tool_used, data_freshness_score, personalization_score, location_accuracy, response_time_sec, rec_quality_score, user_satisfaction, trip_success, time_spent_minutes and would_use_again.

Table 1: Collected Data from APIs

Data Type	Resource	Collected Data	Data Description
Attraction Data	Google	Attraction Name	Name of the attraction
		Attraction Address	Attraction address
		Attraction Rating	The rating of the attraction (1-5)
		User Ratings Total	Total amount rating of the attraction
		Attraction Latitude	Attraction latitude
		Attraction Longitude	Attraction longitude
		Attraction Country	The country of the attraction
		Attraction Type	The type of the attraction, such us tourist attraction, museum and historical place
		Place ID	Attraction ID
Traveler Review	Google	Attraction Name	Name of the attraction
		Attraction Address	Attraction address
		Country	The country of the attraction

		City	The city of the attraction
		User Ratings	Rating of the attraction given from user
		Author Name	Author name
		Place ID	Attraction ID
Weather Date	Open Weather	Date	Date of the weather data
		City	City of the weather data
		Weather	Weather forecast
Flight Data	Amadeus	Flight Number	Flight number
		Airline	Airline company
		Departure Airport	Departure airport
		Arrival Airport	Arrival airport
		Departure Time	Departure time
		Arrival Time	Arrival time
		Ticket Price	Ticket price

Table 2: Parameters for the comparison between Google and ChatGPT

Variable Name	Data Type	Description	Rationale
user_id	Integer	Unique identifier for each user.	To uniquely identify each participant.
age	Integer	Age of the user (18–70).	Age influences technology usage and travel habits.
tech_savviness	Float (1–5)	Level of user's familiarity with technology.	Tech-savvy users may prefer ChatGPT and interact differently.
travel_frequency	Integer	Estimated number of trips per year.	Frequent travelers may seek more recommendations.
query_complexity	Integer (1–3)	Complexity level of the travel query.	Complex queries are more suited to GPT.
tool_used	Integer (0=Google, 1=ChatGPT)	Which tool the user chose to use.	Core comparison variable.
data_freshness_score	Float (1–10)	Freshness and recency of the information.	Recency is crucial for travel relevance.
personalization_score	Float (1–10)	How personalized the recommendations are.	GPT tends to personalize more; personalization increases satisfaction.
location_accuracy	Float (1–10)	Accuracy of location suggestions.	Google has better mapping and location precision.
response_time_sec	Float	Time taken to generate results (in seconds).	Speed of response impacts user experience.
rec_quality_score	Float (1–10)	Overall recommendation quality.	Represents the effectiveness of the recommendation system.

user_satisfaction	Float (1–5)	User satisfaction with the experience.	Key outcome variable representing perceived value.
trip_success	Float (1–10)	Success level of the actual trip.	Evaluates real-world impact of recommendations.
time_spent_minutes	Float	Time user spent interacting with the tool.	Indicates usability and efficiency.
would_use_again	Integer (0=No, 1=Yes)	Whether the user would reuse the same tool.	Measures loyalty and long-term preference.

4 Design Specification

In this paragraph, we will descript every stage of the project including steps of the program and calculation method of synthetic data.

Steps of the program:

User Input:

In this step, users can use natural language provide their travel preference. For instance, they can enter "*I live in Paris and want to go to Okinawa for 3 days travel, start from 20250819, my budget is around 1000 euro, can you help me organize this trip?*".

Using ChatGPT to provide attractions:

User input will send to ChatGPT to obtain related information in JSON format.


```
[
    {
        "Day": 1,
        "Name": "Shurijo Castle",
        "Address": "1 Chome-2 Shurikinjocho, Naha, Okinawa 903-0815, Japan",
        "Latitude": "26.2124",
        "Longitude": "127.7196",
        "Price Level": "Moderate",
        "Availability": "Open",
        "Country": "Japan",
        "Type": "Historical Site"
    },
    {
        "Day": 1,
        "Name": "Kokusai Street",
        "Address": "3 Chome-10-1 Makishi, Naha, Okinawa 900-0013, Japan",
```


Using NLP to extract travel-related information:

Utilizing NLP to extract travel-related information include Departure City, Arrival Country, Arrival City, Travel Period, Travel Days, Budget and Preferences.

Departure City: Paris, Arrival Country: Japan, Arrival City: Okinawa, Travel Period: 2025-08-19 - 2025-08-21, Travel Days: 3 days, Budget: 1000 euros, Preferences: None Attraction Recommendation from

Using Google API to collect attraction data:

Based on arrival city to collect attraction data form Google, including Name, Address, City, Country, Rating, User Ratings Total, Latitude, Longitude, Price Level, Availability, Type, Place_ID.

```
INFO:root:Collecting data for Okinawa with type 'tourist_attraction'
INFO:root:Collecting data for Okinawa with type 'museum'
INFO:root:Collecting data for Okinawa with type 'amusement_park'
INFO:root:Collecting data for Okinawa with type 'aquarium'
INFO:root:Collecting data for Okinawa with type 'zoo'
INFO:root:Collecting data for Okinawa with type 'park'
INFO:root:Collecting data for Okinawa with type 'stadium'
INFO:root:Collecting data for Okinawa with type 'art_gallery'
```


```
[
  {
    "Name": "Okinawa World",
    "Address": "Maekawa-1336 Tamagusuku, Nanjo, Okinawa 901-0616, Japan",
    "City": "Okinawa",
    "Country": "Japan",
    "Rating": 4.3,
    "User Ratings Total": 16771,
    "Latitude": 26.1404807,
    "Longitude": 127.7489447,
    "Price Level": "N/A",
    "Availability": "Available",
    "Type": "tourist_attraction",
    "Place_ID": "ChIJza_GaWlv5TQRy52qkW4L2P4"
  },
  {
    "Name": "Okinawa",
    "Address": "Okinawa, Japan",
    "City": "Okinawa",
    "Country": "Japan",
    "Rating": "N/A",
```


Using open weather API to collect weather data:

Okinawa (20250819 ~ 20250821): 2025-08-19 | broken clouds | Temp: 27.53°C | Humidity: 86% 2025-08-20 | broken clouds | Temp: 27.53°C | Humidity: 86% 2025-08-21 | broken clouds | Temp: 27.53°C | Humidity: 86% No flight data found.

Using hybrid recommendation algorithm provide attraction data:

If the user is new, default to content-based filtering, providing attractions based on Google rating that collected from Google's traveler review. If the user is a return user, using collaborative filtering, recommending attractions based on ratings from similar users.

Using GPT for organizing itinerary:

Put both information from Google and ChatGPT to ChatGPT in order to obtain itinerary. Below is the prompt using in the project.

*****Requirements:*****

- **Attractions**: Select logically grouped attractions by proximity (use latitude/longitude), no duplicates.

- **Daily Schedule**:

- 3–4 activities/day with time slots (e.g., "09:00-10:30").

- Include meal breaks (restaurant name, address, estimated cost).

- **Transportation**: Specify mode (walk/bus/taxi), duration, and cost.

- **Accommodations**: Recommend 1–2 options/night (name, address, € cost).

- **Flights**: Add flight details if data exists (else omit).
- **Weather Adaptation**: Adjust activities based on forecast (e.g., indoor if rainy).

If the user's budget is not enough, still generate a reasonable itinerary based on the most popular attractions and add a field 'notice' in the output JSON with a message for the user, including the estimated total cost ({est_cost}).

The result will be like the format below.


```
"itinerary": [
  {
    "day": 1,
    "date": "2025-08-19",
    "activities": [
      {
        "time": "09:00-10:30",
        "activity": "Visit Okinawa Prefectural Museum and Art Museum (Okimyu)",
        "attraction": {
          "name": "Okinawa Prefectural Museum and Art Museum (Okimyu)",
          "address": "3 Chome-1-1 Omoromachi, Naha, Okinawa 900-0006, Japan",
          "rating": 4.3,
          "latitude": 26.227314,
          "longitude": 127.693849,
          "estimated_cost": "€10"
        },
        "transportation": {
          "mode": "bus",
          "duration": "15 mins",
          "cost": "€2"
        },
        "restaurant": {
          "name": "Restaurant Okimyu",
          "address": "Near Okinawa Prefectural Museum",
          "estimated_cost": "€20"
        }
      },
      {
        "time": "11:00-12:30",
        "activity": "Visit WWII Japanese Okinawa Surrender Site",
        "attraction": {
          "name": "WWII Japanese Okinawa Surrender Site",
          "address": "Morine, Okinawa, 904-0000, Japan",
          "rating": 4.1,
          "latitude": 26.348186,
          "longitude": 127.790504,
```


Calculation method of synthetic data:

Each parameter has different weight and relation with each other; we utilize python to create the dataset based on the method below.

Table 3: Calculation Method of Synthetic Data

Variable Name	Calculation method
user_id	Generated using a sequence from 1 to 1000 *np.arange(1, n + 1)*
age	Random integer between 18 and 70 *np.random.randint(18, 71, n)*
tech_savviness	Inverse relationship to age, plus normal noise; scaled to 1–5 *tech_savviness = np.clip(5 - (age - 18) / 52 * 4 + np.random.normal (0, 0.5, n),1, 5)*
travel_frequency	Poisson distribution influenced by tech_savviness *np.random.poisson (1 + 0.4 * tech_savviness)*
query_complexity	Weighted random choice: 1 (55%), 2 (30%), 3 (15%). *np.random.choice([1, 2, 3], p= [0.55, 0.30, 0.15], size=n)*
tool_used	Probability increases with tech_savviness and query_complexity *p_gpt = np.clip(0.45 + 0.10 * (tech_savviness - 3) + 0.10 * (query_complexity == 3),0.05, 0.95)* *tool_used = np.random.binomial (1, p_gpt)*
data_freshness_score	Higher mean score for Google; random noise added

	mu_fresh = np.where(tool_used == 0, 9, 7) *np.random.normal(mu_fresh, 0.8)*
personalization_score	Higher for GPT, influenced by tech_savvines *mu_pers = np.where(tool_used == 0, 6, 8)* *personalization_score = mu_pers + 0.3 * (tech_savviness - 3) + noise*
location_accuracy	Higher base score for Google, with added noise *location_accuracy = np.clip(np.where(tool_used == 0, 9, 7) + np.random.normal(0, 0.6, n), 1, 10)*
response_time_sec	Normally distributed around 5s for Google, 6s for GPT *np.where(tool_used == 0, normal (5, 1.2), normal (6, 1.4))*
rec_quality_score	Weighted sum of freshness, personalization, and location scores; penalized for complex queries with Google *rec_quality = 0.35*freshness + 0.35*personalization + 0.3*location - 0.4*(complexity-1)* *rec_quality -= 0.2*(complexity-1) * (tool_used == 0)*
user_satisfaction	Function of recommendation quality, response time, and tech_savviness *satisfaction = 1 + 0.4 * rec_quality - 0.05 * response_time + 0.12 * (tech_savviness - 3) + noise*
trip_success	Driven by satisfaction, travel frequency, and query complexity *trip_success = 4 + 0.6 * satisfaction + 0.25 * travel_frequency - 0.25 * (complexity - 1) + noise*
time_spent_minutes	Google users spend more time (normal distribution), GPT users spend less *Google: normal(25, 10) ; GPT: normal(15, 5)*
would_use_again	Binary outcome based on satisfaction + quality score > threshold *would_use_again = (rec_quality_score + user_satisfaction > 6.5).astype(int)*

After generating the dataset, we used python to create a correlation matrix to figure out the relation between each parameter. When making it to a real system, we would know which part should be more focused to increase user friendly of the system.

5 Implementation

In this project we use python to build the travel recommendation chatbot system, which provide users an interface to interact. After receiving user input, the program in the backend will start working to extract keywords and collecting related information. In the end, ChatGPT will be used to organize the itinerary and provide to user in the frontend. In this system, multiple APIs were used to collect travel related data, including attraction data, traveller review, weather data and flight data.

Attraction Data
API: Google Places API
Program: Attractions.py
Programming language: python
Integrated libraries:

Table 4: Library used in Attraction.py

Library	Purpose
requests	Sends HTTP GET requests to Google Places API.
logging	Records info, errors, and debugging output in a readable format.
time	Introduces delays (e.g., for API rate limiting or waiting for next_page_token).
urllib.parse	Encodes query strings (e.g., museum in Paris) into safe URL format.
sys	Reads command-line arguments to allow dynamic city input.
json	Saves collected data into a .json file.

Description:

We collect attraction data from Google API. In this program, user can input their arrival city for collecting tourist spots in the city. The type of attractions we collect in this program includes "tourist_attraction", "museum", "amusement_park", "aquarium", "zoo", "park", "stadium", "art_gallery", "church", "mosque", "hindu_temple", "synagogue", "library", "casino", "night_club", "movie_theater", "bowling_alley", "campground", "natural_feature","shopping_mall","historical_place","point_of_interest","establishment" and "landmark".

When collecting data, we use *quote(f"{place_type} in {city}")* to do it, such as museum in Paris. After collecting all the data, this program will save it as a JSON file for furthering analysis by the recommendation system.

Traveler Review Data

API: Google Places API
Program: Traveler_Review.py
Programming language: python
Integrated libraries:

Table 5: Library used in Traveler_Review.py

Library	Purpose
requests	Sending HTTP GET requests to Google Places API
logging	Logging info, errors, and debugging output
time	Handling delays (e.g., between paginated requests or retries)
pandas	Storing, manipulating, and exporting review data
urllib.parse	URL-encoding user inputs or search queries

Description:

We collect travel review data from Google place API. This data we used as a database for judging if the user is new or return. When providing suggestion to users, we will use the attraction rate from this dataset as a definition to recognize the attraction is popular or not.

Weather Data

API: OpenWeatherMap API
Program: Weather_data.py
Programming language: python
Integrated libraries:

Table 6: Library used in Weather_data.py

Library	Purpose
requests	For sending HTTP requests to the weather API
json	For handling JSON parsing
pandas	To organize and export data to Excel
datetime, timedelta	For calculating date ranges and formatting dates

Description:

We collect weather data from OpenWeatherMap API. The input of this program is the destination and date from travellers' input data. The program will connect with OpenWeatherMap API for collecting needed information and provide for users in the recommendation system.

Flight Data
API: Amadeus API
Program: Flight_data_Amaduse.py
Programming language: python
Integrated libraries:

Table 7: Library used in Flight_data_Amaduse.py

Library	Purpose
amadeus	Official SDK for Amadeus travel APIs (flight search, hotels, etc.)
pandas	To format, manipulate, and save data (to JSON/Excel/CSV, etc.)
sys.path.append	Allows importing your custom module (airport_iata_lookup.py)

Description:

Flight data we use Amadeus API for collecting. The input of this program is the IATA code of departure and arrival city as well as departure date. With the help of this API, we can get flight information for ChatGPT to combine in the itinerary when organizing the data.

ChatGPT has been using to collect attraction data in the beginning and organize all the collected travel data to generate itinerary in the end.

Attraction Data from ChatGPT
API: OpenAI
Program: travel_info_analyzer.py
Programming language: python
Integrated libraries:

Table 8: Library used in travel_info_analyzer.py

Library	Purpose
openai	Communicating with OpenAI's ChatGPT models
json	Parsing and writing JSON
re	Regular expressions for extracting valid JSON arrays from raw text

Description:

We use ChatGPT for collecting attraction data as well. The prompt we use in this program is "

Extract the following travel information from the user input: departure country, departure city, arrival country, arrival city, travel date range, travel days, budget (euros), and preferences.

Based on these, suggest tourist attractions for each travel day (number of days = travel days) in JSON array format.

Each day should have **three** attractions, and the output should be a single flat JSON array (not grouped by day), with each object containing:

{{
"Day": <day number, integer>,
"Name": "",
"Address": "",
"Latitude": "",
"Longitude": "",
"Price Level": "",
"Availability": "",
"Country": "",
"Type": ""
}}
User input: {user_input}

Respond ONLY with a valid JSON array in the format: [{{}},{{}},...], with NO extra text, explanation, or markdown. Only output the JSON array."

The result of this output will combine with the attraction data offered from Google Place API in the end and send to ChatGPT for organizing itinerary.

During the recommendation process, we use hybrid recommendation algorithm to give user suggestion based on their preference.

Provide attraction recommendations collected from Google

Program: Recommender_Analysis.py
Programming language: python
Integrated libraries:

Table 9: Library used in Recommender_Analysis.py

Library	Purpose
pandas	Data loading, cleaning, transformations and pivot tables.
TfidfVectorizer	Convert textual features (here: combined Type + Country) into numeric vectors for content-based similarity.
cosine_similarity	Compute similarity between vectors
datetime / timedelta	Parse travel date strings and generate per-day ranges.

Description:

The recommendation system will recognize the identity of system users, if the user is new, default to content-based filtering; however, if the user is a return user, using collaborative filtering. The attraction given to travelers are chosen by their rating from Traveler Review dataset.

```python
def hybrid_recommendation(traveler_df, attraction_df, user_id, city_or_country, travel_duration, top_
    """
    Combine content-based and collaborative filtering for hybrid recommendations.
    If the user is new, default to content-based filtering.
    Recommend 5 unique attractions per day, no repeats.
    """
    # Filter attractions by location
    #filtered_attractions = filter_attractions_by_location(attraction_df, city_or_country)

    # Check if the user is new
    if user_id not in traveler_df['Author_Name'].unique():
        # Default to content-based filtering for new users
        recommendations = content_based_filtering(attraction_df, top_n=len(attraction_df))
    else:
        # Get collaborative filtering recommendations
        cf_recommendations = collaborative_filtering(traveler_df, attraction_df, user_id, top_n=len(a

        if not cf_recommendations.empty:
            # Get content-based recommendations for the top collaborative recommendation
            cb_recommendations = content_based_filtering(attraction_df, cf_recommendations.iloc[0]['P

            # Combine and deduplicate recommendations
            recommendations = pd.concat([cf_recommendations, cb_recommendations]).drop_duplicates(sub
        else:
            # If no collaborative recommendations, default to content-based filtering
            recommendations = content_based_filtering(attraction_df, top_n=len(attraction_df))
```

As for analyzing synthetic data, we used python as well, to define the relationship of each parameter, we create a correlation matrix to identify the corelation between each other.

Create Synthetic Data
Program: Analysis_synthetic_data.py
Programming language: python
Integrated libraries:

Table 10: Library used in Analysis_synthetic_data.py

Library	Purpose
NumPy	Random draws, numeric computations, vectorized math. Use for efficient synthetic-data generation.
pandas	Assemble DataFrame, I/O to CSV/Excel, data manipulation.
matplotlib.pyplot & seaborn	Plotting the correlation heatmap and saving figure.

Description:

In this program, the variables will calculate by the weight that given based on their specific feature. For instance, Google's location accuracy should higher than ChatGPT due to the accurate geography data it has.

Backend
Program: chat_backend.py
Programming language: python
Integrated libraries:

Table 11: Library used in chat_backend.py

Library	Purpose
Flask	HTTP server exposing /chat endpoint.

flask_cors.CORS	Enables CORS for local front-end dev (localhost:8000).

Description:

This program is used to communicate with all other programs and organize the output to provide to frontend of the system.

Frontend
Program: chatbot.html
Programming language: python
Description:

This program is a chatbot interface for user to communicate with the recommendation system.

Organize itinerary by ChatGPT
Program: Itinerary_organize.py
Programming language: python
Integrated libraries:

Table 12: Library used in Itinerary_organize.py

Library	Purpose
pandas	Used to load and manipulate travel-related data such as attractions, flights, and weather from JSON files. Help filter, sort, and process the data before passing it to the prompt.
datetime	Used to parse, convert, and compute date ranges for the travel duration, helping generate dates for the itinerary.
time	Used for delay/sleep between retries on API connection errors.
json	Used for reading JSON data files and parsing the OpenAI API response.
argparse	Parses command-line arguments such as travel duration, budget, and user preference for filtering and generating the itinerary.
re	Used to extract JSON from the raw string returned by the OpenAI API, ensuring the program can parse the valid JSON content.
openai	Used to send a prompt containing travel data to OpenAI's GPT-4 model, which generates a detailed travel itinerary in JSON format.

Description:

This program utilizes ChatGPT to organize all the collected travel related information into an itinerary based on users' inputs.

6 Evaluation

In the following is the description of correlation matrix and the conclusion of this analysis.

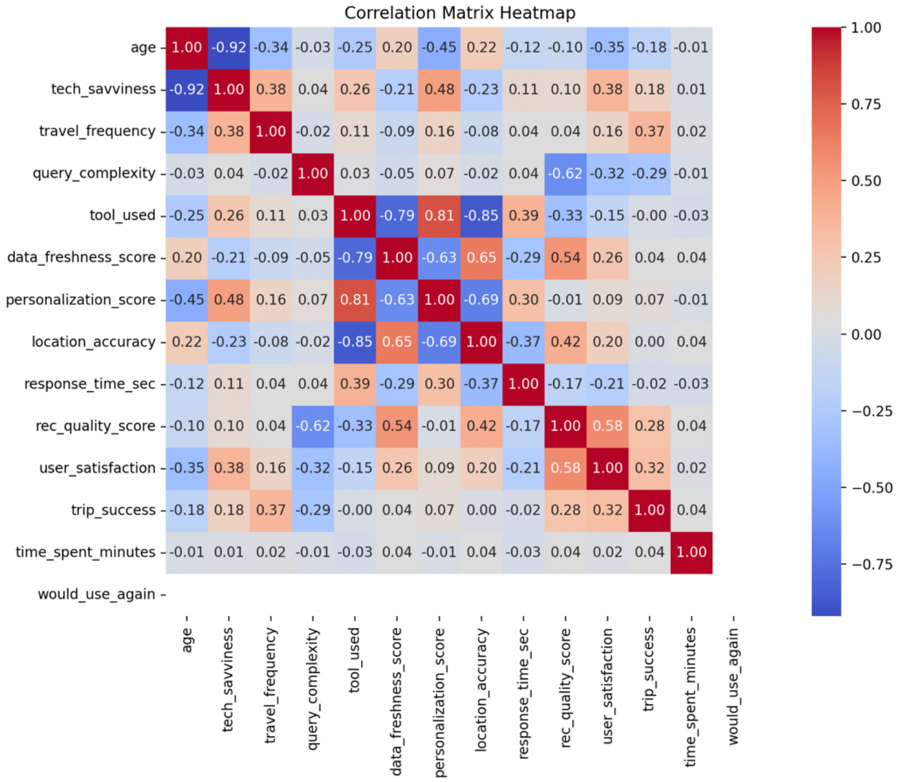

Figure 4: Correlation Metrix

In the correlation matrix we can find the relation between each parameter, which would describe below.

Table 13: Correlation between Google and ChatGPT

Variable Pair	Correlation
tech_savviness vs age	-0.92
tool_used vs personalization_score	0.81
tool_used vs location_accuracy	-0.85
rec_quality_score vs user_satisfaction	0.58
tool_used vs response_time_sec	0.39
trip_success vs user_satisfaction	0.58
user_satisfaction vs would_use_again	0.32
query_complexity vs rec_quality_score	-0.62

6.1 tech_savviness and age

The correlation between tech_savviness and age has very strong negative relation, -0.92. Which means when people getting older, they could have more difficult to adapt technology. On the contrast, younger generation are better at technology than elder group. Therefore, these two parameters show negative correlation between each other.

6.2 tool_used and personalization_score

The correlation between tool_used and personalization_score has positive relation, 0.81. When users require more personalization result, ChatGPT can do it better than traditional Google search. As we mention before, ChatGPT can understand human language and based on this provide an output that closer to users' preference.

6.3 tool_used and location_accuracy

The correlation is -0.85 for tool_used and location_accuracy. Which means users likely rely on fewer tools since the chosen tool can provide right location information. Usually, Google search could provide more accuracy result than ChatGPT due to the misunderstanding that ChatGPT might has.

6.4 rec_quality_score and user_satisfaction

rec_quality_score and user_satisfaction have positive correlation, 0.58. The higher recommendation quality can lead to higher user satisfaction.

6.5 tool_used and response_time_sec

The correlation between tool_used and response_time_sec is positive, 0.39. That means response time can influence users' decision for choosing tools.

6.6 trip_success and user_satisfaction

The correlation between trip_success and user_satisfaction is positive, 0.58. Which means the higher user satisfaction when the travel plan is good and successful.

6.7 user_satisfaction vs would_use_again

The correlation between user_satisfaction and would_use_again is positive, 0.32. When the chosen tool can provide better response for users, they would have higher active to reuse it again.

6.8 query_complexity vs rec_quality_score

query_complexity vs rec_quality_score have negative correlation. Google search could provide inaccurate output when the input query is too complicate; however, due to the understanding of natural language, ChatGPT could provide better response than Google when the input data is too complicate.

6.9 Discussion

This section provides a critical discussion of the main findings from the eight experiments described before.

Overall, the result shows the relation between user characteristics, tool choice, and satisfaction levels. Noticeably, tech_savviness and age have strong negative correlation (-0.92), which means the younger generation are more comfortable with technology than older users. The correlation between tool_used and personalization_score (0.81) has very strong positive with each other. Which means that ChatGPT tends to provide more personalized results compared to traditional Google search. This align with the previous research that the theoretical understanding of large language models being better at realizing human language intent. On the contrast, the relationship between tool_used and location_accuracy showed a negative correlation (-0.85), reflecting that Google search may still be more reliable in location-specific queries due to the integration with geospatial data services.

The correlations between rec_quality_score and user_satisfaction (0.58) as well as trip_success and user_satisfaction (0.58) both confirm that the recommendation and successful trip outcomes positively influence how user feel about the system. However, the correlation between user_satisfaction and would_use_again (0.32) has only slightly positive relation, which could mean that satisfaction alone may not fully determine users' willingness to reuse a tool, the considerate of trust, familiarity, or convenience maybe necessary as well.

7 Conclusion and Future Work

With the ever progressing of technology, people can obtain travel information by utilizing Google search, travel related websites and LLMs. Compared with the traditional way that requires the help of travel agencies and guidebooks, it is easier for travellers collecting the information they need nowadays. Moreover, people in increasing numbers tend to choose independent trip rather than book an itinerary from travel agent. However, the trip information collection is time consuming and tiring sometimes. People would need to search attraction data, flight data, hotel data, weather information and so on. If using traditional Google search could consume many times and energy for planning a trip.

The advent and advance of LLMs such as ChatGPT provide travelers another tool for trip planning. With the ability of understanding natural language, LLMs can output the response that fit users' multiple preference and interacting with them like a real human agent. However, LLMs could lack real-time data and having hallucination sometimes.

In this study, we combine data from multiple APIs for real-time data and ChatGPT to create a travel related chatbot that can provide user information about their trip and help them plan the itinerary that align with their preference. To evaluation the system, we create a synthetic dataset which include 15 variables to analysis the efficiency and practically. In the end, we create a correlation matrix to analysis the correlation between each parameter and use these results to design the system in the future to increase user satisfaction.

Additionally, the dataset used in this study may have limitations in terms of sample size, demographic balance, and tool familiarity. In the further work, we should consider larger and more diverse sample groups, as well as alternative experimental designs to create a more comprehensive evaluation as well as upgrade the travel chatbot to provide a better service for travelers.

References

UN Tourism, 2025. "International tourism recovers pre-pandemic levels in 2024." UN Tourism, Available at: https://www.unwto.org/news/international-tourism-recovers-pre-pandemic-levels-in-2024 [Accessed on 9 February 2025].

TTN, 2024"【2025 年旅遊趨勢大揭密】自由行 VS 團體遊 旅行社如何搶客？", Available at: https://www.ttnmedia.com/?p=110656 [Accessed on 9 February 2025].

Angelina Villa-Clarke, 2024. "Travel Trends Report 2025: Solo Travel." Forbes, Available at: https://www.forbes.com/sites/angelinavillaclarke/2024/11/27/travel-trends-report-2025-solo-travel/ [Accessed on 9 February 2025].

Totallydublin, 2024. " Solo Travel Trends Shaping Individual Experiences Globally." Available at: https://www.totallydublin.ie/more/solo-travel-trends-shaping-individual-experiences-globally/ [Accessed on 9 February 2025].

Pruvo, 2025. " The Rise of Personalized Travel Experiences: Tailoring Offerings for Customers." Available at: https://www.pruvoai.com/post/the-rise-of-personalized-travel-experiences-tailoring-offerings-for-customers [Accessed on 9 February 2025].

Dana Dunne, 2022. "The Future Of Personalization In The Travel Industry." Forbes, Available at: https://www.forbes.com/sites/danadunne/2022/01/27/the-future-of-personalisation-in-the-travel-industry/ [Accessed on 9 February 2025].

Shreeansh Mishra, Mansurali Anifa, and Jigisha Naidu, 2025. "AI-Driven Personalization in Tourism Services: Enhancing Tourist Experiences and Business Performance." Integrating Architecture and Design Into Sustainable Tourism Development, Available at: https://www.igi-global.com/chapter/ai-driven-personalization-in-tourism-services/366717 [Accessed on 9 February 2025].

Alotaibi, R., Ali, A., Alharthi, H. & Almehamdi, R., 2020. AI Chatbot for Tourist Recommendations: A Case Study in the City of Jeddah, Saudi Arabia.," International Association of Online Engineering.

Hai Thanh Nguyen, Thien Thanh Tran, Phat Tan Nham, Nhi Uyen Bui Nguyen, Anh Duy Le, 2023. "AI Chatbot for Tourist Recommendations: A Case Study in Vietnam", College of Information and Communication Technology, vol. 28, no. 2, pp. 232–244

Cole Stryker and Jim Holdsworth, 2024. "What is NLP (natural language processing)? "IBM, Available at: https://www.ibm.com/think/topics/natural-language-processing [Accessed on 24 February 2025].

Meira, J., Carneiro, J., Bolón-Canedo, V., Alonso-Betanzos, A., Novais, P., & Marreiros, G., 2022. "Anomaly Detection on Natural Language Processing to Improve Predictions on Tourist Preferences. " Electronics, 11(5), 779.

Byron Jaramillo, Edison Loza-Aguirre, Luis Terán, 2023. "Designing a Framework for Explainable Health Recommender System Based on the Ecuadorian Data Protection Regulations", 2023 Ninth International Conference on eDemocracy & eGovernment (ICEDEG), pp.1-8

M. S. Roobini, S. P. David, S. Lokeshwaran, V. E and A. D, 2024. "Content Filtering Based Navigation Recommendation System Using NLP," 2024 Ninth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), Chennai, India, pp. 1-8

K. Binabdullah and N. Tongtep, 2021. "Comparative Study on Natural Language Processing for Tourism Suggestion System," 2021 36th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Jeju, Korea (South), pp. 1-4

Akhilesh Kumar Sharma, Bhavna Bajpai, Rachit Adhvaryu, Suthar Dhruvi Pankajkumar, Prajapati Parthkumar Gordhanbhai, Atul Kumar, 2023. " An Efficient Approach of

Product Recommendation System using NLP Technique, " Volume 80, Part 3, Pages 3730-3743,

A. Dixit, K. Gupta and D. G, 2023. "Movie Recommendation Chatbot Using Cosine Similarity and NLP tools," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, pp. 1-7

Komal Londhe, Nikita Dharmadhikari, Parth Zaveri, Unal Sakoglu, 2024. "Enhanced Travel Experience using Artificial Intelligence: A Data-driven Approach", Procedia Computer Science, Volume 235, Pages 1920-1928

Lin K, Yang S, Na SG., 2022. "Collaborative Filtering Algorithm-Based Destination Recommendation and Marketing Model for Tourism Scenic Spots. " Computational Intelligence and Neuroscience, Volume 2022, Article ID 7115627, 7 pages

Nan X, Kayo Kanato, Wang X., 2022. "Design and Implementation of a Personalized Tourism Recommendation System Based on the Data Mining and Collaborative Filtering Algorithm. " Computational Intelligence and Neuroscience, Volume 2022, Article ID 1424097, 14 pages

Cheng, X., 2021. "A travel route recommendation algorithm based on interest theme and distance matching. " EURASIP J. Adv. Signal Process. 2021, 57

N. A. L. Nethmin, L. G. S. Saranga, A. D. T. Dilshan, H. E. Wijewickrama, N. H. P. Ravi Supunya Swarnakantha and U. U. Samantha Rajapaksha, 2023. "Enhancing Tourism Experience in Sri Lanka through Data Driven Recommendations," 2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), pp. 1-7

A. Vysotsky et al., 2019. "Online Tourism System for Proposals Formation to User Based on Data Integration from Various Sources," 2019 IEEE 14th International Conference on Computer Sciences and Information Technologies (CSIT), pp. 92-97

Rina Diane Caballar and Cole Stryker, 2024. " What is a recommendation engine?" IBM, Available at: http://ibm.com/think/topics/recommendation-engine [Accessed on 13 March 2025].

Nikas, I.A., Koutras, A., Theodora, A., 2025. "A Conceptual Framework for Developing and Evaluating Personalized Tourist Recommendation Systems Using Large Language Models. " Innovation and Creativity in Tourism, Business and Social Sciences. IACuDiT 2024. Springer Proceedings in Business and Economics. Springer, Cham.

Harmanpreet Singh, Nikhil Verma, Yixiao Wang, Manasa Bharadwaj, Homa Fashandi, Kevin Ferreira, and Chul Lee. 2024. "Personal Large Language Model Agents: A Case Study on Tailored Travel Planning. " In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track, pages 486–514, Miami, Florida, US. Association for Computational Linguistics

S. Ge, P. Ye, R. Zhang, M. Zhou, H. Dong and F. -Y. Wang, 2025. "LLM-Driven Cognitive Modeling for Personalized Travel Generation," in IEEE Transactions on Computational Social Systems

Dongjie Yang, Chengqiang Lu, Qimeng Wang, Xinbei Ma, Yan Gao, Yao Hu, Hai Zhao, 2025. "Plan Your Travel and Travel with Your Plan: Wide-Horizon Planning and Evaluation via LLM." Shanghai Jiao Tong University

Tianming Liu, Manzi Li, Yafeng Yin, 2025. "Aligning LLM with human travel choices: a persona-based embedding learning approach." 1Department of Civil and Environmental Engineering, University of Michigan.

Song, S., Yang, C., Xu, L., Shang, H., Li, Z., & Chang, Y., 2024. "TravelRAG: A Tourist Attraction Retrieval Framework Based on Multi-Layer Knowledge Graph." ISPRS International Journal of Geo-Information, 13(11), 414.

Y. Chen et al., 2025. "Toward Interactive Next Location Prediction Driven by Large Language Models." in IEEE Transactions on Computational Social Systems, Page: 1 – 17

Aribas, E., & Daglarli, E., 2024. "Transforming Personalized Travel Recommendations: Integrating Generative AI with Personality Models. " *Electronics*, *13*(23), 4751.

Ahmad Hassan Afridi, Ansar Ul Haque Yasar, Tarek R. Sheltami, 2025. "Serendipity and LLM-based Recommender System for Smart Transportation", Transportation Research Procedia,Volume 84, Pages 251-258, ISSN 2352-1465

Banerjee, A., Satish, A., Wörndl, W., 2025. "Enhancing Tourism Recommender Systems for Sustainable City Trips Using Retrieval-Augmented Generation. " Recommender Systems for Sustainability and Social Good. RecSoGood 2024 Communications in Computer and Information Science, vol 2470. Springer, Cham.

T. Alenezi and S. Hirtle, 2022. "Normalized Attraction Travel Personality Representation for Improving Travel Recommender Systems," in *IEEE Access*, vol. 10, pp. 56493-56503

Z. Zhang, 2024. "Evaluation of the Effect of Tourism Recommendation System based on AI," 2024 3rd International Conference on Artificial Intelligence and Computer Information Technology (AICIT), Yichang, China, pp. 1-4

Mohamed Badouch & Mehdi Boutaounte, 2023. "Personalized Travel Recommendation Systems: A Study of Machine Learning Approaches in Tourism." Journal of Artificial Intelligence Machine Learning and Neural Network, ISSN: 2799-1172, Vol: 03, No. 03, April-May 2023

K. A. Fararni, F. Nafis, B. Aghoutane, A. Yahyaouy, J. Riffi and A. Sabri, 2021. "Hybrid recommender system for tourism based on big data and AI: A conceptual framework," in Big Data Mining and Analytics, vol. 4, no. 1, pp. 47-55

S. Aggrawal, D. Teotia, Y. Yash and S. Shukla, "Travel Itinerary Generation using AI and Recommendation Systems, 2025. " *2025 2nd International Conference on Computational Intelligence, Communication Technology and Networking (CICTN)*, Ghaziabad, pp. 498-502