International Journal of Computer Techniques–IJCT Volume 12 Issue 6, November 2025

Open Access and Peer Review Journal ISSN 2394-2231 https://ijctjournal.org/

Personalized Travel Itinerary Generation Using Large Language Models and Generative AI

Himani Rajput
Department of Information Technology
MMMUT Gorakhpur, India
2022071037@mmmut.ac.in

Prabhat Yadav
Department of Information Technology
MMMUT Gorakhpur, India
2022071049@mmmut.ac.in

Vishal Tiwari
Department of Information Technology
MMMUT Gorakhpur, India
2022071077@mmmut.ac.in

Prachi Verma
Assistant Professor
Department of Information Technology
MMMUT Gorakhpur, India
prachi.verma1499@gmail.com

***Abstract*— The rapid evolution of digital tourism platforms has demonstrated the pressing need for automated systems that can transform fragmented travel information into cohesive and highly personalized itineraries. While traditional systems offer isolated recommendations, users increasingly expect detailed day-wise plans that reflect real travel behavior, practicality, and comfort. The complexity of modern travel decision-making—shaped by cost fluctuations, diverse preferences, and varying destination characteristics—demands solutions capable of synthesizing multiple factors simultaneously. The system discussed in this research addresses these expectations by structuring user requirements into logical sequences that mirror the planning style of experienced human travel coordinators. This enhanced capability results in itineraries that feel both intuitive and realistic, bridging the gap between raw data and meaningful travel experiences.Unlike conventional recommendation platforms that provide isolated suggestions, the system emphasizes the creation of complete, day-wise itineraries that reflect realistic travel behavior and geographical flow. It aims to reduce planning fatigue and improve decision-making efficiency by ensuring that suggested activities align with human comfort levels, time availability, and practical feasibility. The methodology prioritizes organization, clarity, and sequencing, ensuring that travelers receive itineraries that not only highlight important attractions but also provide a balanced distribution of activities across the entire trip.**

Keywords*— Artificial Intelligence, Large Language Models, Travel Recommendation Systems, Personalization, FastAPI, Generative AI.

I. Introduction

Tourism has undergone a significant transformation, with travelers increasingly seeking autonomy and personalization in planning. Unlike earlier decades where travel agencies dominated trip planning, modern travelers rely heavily on digital resources such as travel blogs, review sites, maps, and social media recommendations. While these platforms provide a rich set of choices, they lack the ability to synthesize information into coherent, easy-to-follow itineraries. As a result, individuals spend a substantial amount of time researching, comparing, and organizing activities to match their personal interests, available time, and travel style.

The complexity of travel planning arises not from a lack of information but from an excess of fragmented data scattered across multiple sources. A traveler may find hotel suggestions on one platform, must-visit attractions on another, and local transportation options on yet another. Connecting these elements into a logical sequence becomes mentally taxing, especially for multi-day or multi-city trips. Moreover, travelers value balance—ensuring the plan includes not only sightseeing but also adequate rest, meal breaks, and travel buffer time.

The difficulty lies not in accessing information, but in organizing it in a way that aligns with individual travel styles. A planned trip must respect personal preferences, daily energy levels, destination characteristics, and logistics such as transportation and opening hours. Yet, manually arranging these elements—especially for multi-day trips—requires considerable time, effort, and familiarity with the destination. First-time travelers often struggle the most, as they lack knowledge about local distances, peak hours, optimal visit timings, and realistic time allocations. Such gaps lead to poorly structured itineraries that may be impractical, rushed, or unbalanced.

The evolving expectations of modern travelers further highlight the need for structured planning assistance. Travelers now seek personalized experiences rather than generic lists of sites. They value plans that reflect their interests, whether cultural, recreational, adventurous, historical, or nature-focused. Likewise, factors such as group composition, age considerations, budget levels, and travel pace require thoughtful integration into the itinerary-building process. An itinerary meant for a solo traveler may not suit a family with elderly members, while a high-budget traveler may expect premium experiences and comfort-oriented suggestions.

Given these complexities, the challenge is not the lack of recommendations but the absence of a structured mechanism that organizes relevant choices into a logical, sequential plan. A well-designed itinerary must ensure smooth transitions between activities, avoid unnecessary travel loops, allocate appropriate break times, and provide a balanced distribution of attractions throughout the trip. The primary aim of structured itinerary generation is to reduce planning stress and provide travelers with a clear, organized, and user-friendly travel blueprint.

The focus of this study is to analyze and develop an itinerary-generation system that translates user inputs into well-structured, day-by-day travel schedules. The system's approach emphasizes clarity, coherence, and user-centric design, ensuring that the resulting itinerary reflects both practicality and personalization. By addressing the limitations of unstructured travel data, the system contributes to simplifying the planning process, improving travel experiences, and empowering users with a reliable, ready-to-follow travel plan tailored to their unique preferences.

This paper presents the design and implementation of an *AI-based Itinerary Planner.* The primary contributions of this study are:

1. A robust architecture for converting unstructured user inputs into structured travel plans.
2. Integration of Generative AI (Gemini) with a lightweight backend (FastAPI) for real-time generation.
3. Evaluation of the system's ability to handle constraints like budget and specific interests.

II. LITERATURE REVIEW

In recent years, research in tourism technology has increasingly focused on crafting systems capable of providing meaningful, context-aware travel support. Traditional filtering methods such as Collaborative Filtering and Content-Based Filtering laid foundational work for recommendation systems but proved inadequate in generating complete itineraries. These methods primarily offer isolated suggestions rather than interconnected plans. For instance, recommending five popular attractions in a city does not inherently provide guidance on which to visit first, how long each visit should take, or whether they are realistically compatible within a single day of travel.

Scholarly work also emphasizes the importance of contextual decision-making in tourism planning. Destinations vary greatly based on peak seasons, local events, cultural festivities, or even weekly closing schedules. These dynamic factors influence travel behavior and can drastically alter a visitor's experience. Researchers highlight that travelers often rely on subjective elements—such as crowd levels, comfort, ambience, and personal pace—making experiential data equally important as factual information.

Another widely studied area is the integration of social feedback in travel decision-making. Online reviews, community posts, and travel storytelling provide valuable insights into traveler experiences, sentiment, and satisfaction. These real-human perspectives reveal practical challenges such as long queues, limited transportation options, or unexpected closures—factors that play a crucial role in shaping realistic itineraries.

Across all existing research, a consistent gap appears: although there is abundant work on generating recommendation lists, relatively fewer studies offer solutions that produce structured itineraries. This highlights a clear need for systems that combine recommendation accuracy with day-wise organization, contextual awareness, and real-world feasibility.

A. Traditional Recommendation Systems in Tourism

Early research in e-tourism was dominated by two primary filtering paradigms: Collaborative Filtering (CF) and Content-Based Filtering (CBF).

- *Collaborative Filtering*: As noted by Gavalas et al. [1], CF algorithms, such as Matrix Factorization and Nearest Neighbor, operate on the assumption that users who agreed in the past will agree in the future. While effective for recommending single items (e.g., a specific hotel or museum), CF suffers significantly from the "Data Sparsity" problem. In the tourism domain, users visit only a tiny fraction of available destinations, leading to sparse user-item interaction matrices that degrade recommendation quality. Furthermore, CF is plagued by the "Cold Start" problem, where the system fails to provide relevant suggestions for new users with no prior booking history.
- *Content-Based Filtering*: CBF approaches attempt to solve this by matching user profiles with item attributes (e.g., matching a user's interest in "History" with sites tagged "Monument"). However, these systems are often limited by "Over-Specialization," where the user is restricted to seeing items similar to what they have already interacted with, preventing the discovery of novel experiences. Crucially, neither CF nor CBF can inherently generate a *sequential* itinerary; they provide a list of isolated spots without considering logistical constraints like travel time between locations or opening hours.

B. Predictive AI and Deep Learning Models

To address the sequential nature of travel, researchers introduced Deep Learning (DL) models. Specifically, Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks became popular for "Next Point-of-Interest" (POI) prediction.

- *Sequential Pattern Mining*: Studies [2] have demonstrated that LSTMs can effectively analyze a tourist's historical trajectory to predict their next likely stop. For instance, if a tourist visits the Eiffel Tower, an LSTM model might predict the Louvre Museum as the next stop based on crowd flow patterns.
- *Limitations*: While superior to static filtering, these predictive models operate as "Black Boxes." They lack explainability—often failing to inform the user *why* a specific sequence was chosen. More importantly, they are *discriminative* rather than *generative*. They classify or predict existing data points but cannot synthesize a net-new plan from scratch based on abstract, unstructured constraints (e.g., "Plan a romantic but budget-friendly trip for an elderly couple"). They rely heavily on massive, structured datasets of user trajectories, which are often unavailable or privacy-sensitive.

C. Generative AI and Large Language Models (LLMs)

The emergence of Transformer-based architectures has marked a paradigm shift from "Information Retrieval" to "Information Synthesis." Large Language Models (LLMs), such as GPT-4 and Google Gemini, utilize massive training corpora to perform "Zero-Shot" and "Few-Shot" reasoning.

- *Semantic Reasoning***:** Unlike traditional search engines that perform keyword matching, LLMs employ self-attention mechanisms to understand complex, multi-faceted user queries. Li et al. [3] highlight that LLMs can process natural language constraints that are difficult to encode mathematically, such as "vibe," "pace of travel," or "cultural immersion."
- *Automated Planning Agents:* Recent work [4] explores the use of LLMs as autonomous agents. By acting as a reasoning engine, the LLM can decompose a complex goal ("Plan a 3-day trip") into sub-tasks (Day 1 planning, logistics check, budget balancing). This paper builds upon this domain by wrapping the raw LLM in a structured application layer. This mitigates the common "Hallucination" issue—where LLMs invent non-existent places—by enforcing strict output schemas (JSON) and grounding the generation in a predefined prompt structure. This approach combines the creative flexibility of Generative AI with the structural rigidity required for actionable software applications.

III. System Methodology

The system methodology is designed to ensure that the transformation of user requirements into structured, day-by-day travel itineraries occurs in a coherent, organized, and logically consistent manner. This methodology focuses on clarity, modularity, and practicality so that each stage contributes meaningfully to the accuracy and usability of the final itinerary. The overall design follows a layered approach, where each component performs a dedicated function, ensuring the entire workflow remains streamlined and adaptable for future improvements.

A. *User Input Collection and Processing*

The first stage involves capturing essential travel-related inputs from users. These inputs form the foundation of the itinerary-generation process. Travelers typically provide destination details, number of days, preferences, budget range, age group, and type of activities they are interested in. The methodology ensures that even vague or loosely stated preferences are categorized meaningfully. For instance, if a user selects “nature,” the system maps this preference to related activities such as lakes, gardens, beaches, or scenic viewpoints. This categorization lays the groundwork for accurate filtering and grouping in subsequent steps.

In this phase, user inputs are validated to ensure completeness and avoid contradictions. Incomplete or inconsistent entries—such as selecting a short duration but requesting an unusually large number of places—are interpreted intelligently. The system prompts for clarification if necessary or resolves ambiguity using predefined logic. The goal is to achieve well-structured input data that can be processed smoothly across the system.

B. *Preference Interpretation and Classification*

Once input data is validated, the next step is to classify the user’s travel style. Preferences differ widely across individuals: some tourists prefer activity-heavy schedules filled with exploration, while others prioritize rest, comfort, and slow travel. Similarly, family groups may require balanced activities suitable for all ages, while solo travelers may be open to extended walking or adventure-based experiences.

This stage analyzes the user’s inputs to determine:

- **Pace of travel** (relaxed, moderate, fast)
- **Category of interest** (culture, history, nature, food, leisure, adventure)
- **Budget sensitivity**
- **Accessibility constraints**
- **Time-of-day preferences**

By understanding these aspects, the methodology ensures that itinerary construction aligns with user expectations. For instance, a relaxed traveler receives more free time and fewer back-to-back activities, while a fast-paced traveler receives a fuller schedule.

C. *Attraction Selection and Filtering*

The attraction-selection module filters relevant points of interest based on the user’s preferences. It excludes irrelevant categories and prioritizes spots that best match the chosen theme of the trip. This filtering process also considers factors such as:

- Distance from the user’s accommodation
- Typical weather dependence of specific attractions
- Crowd tendencies during certain hours
- General cost categories (aligned with user budget)

The goal is not to include as many attractions as possible but to select those that provide value, coherence, and feasibility within the trip duration. Unnecessary or redundant attractions are removed to prevent overwhelming the user with an unrealistic schedule.

D. *Day-Wise Grouping and Logical Sequencing*

This is the core of the methodology and the most important stage. The system ensures that selected attractions are arranged into day-wise blocks that reflect realistic travel flow.

1) *Geographical Clustering*

Attractions located close to each other are grouped on the same day to minimize travel time. This reduces fatigue and prevents users from making long, unnecessary trips between activities.

2) *Time Allocation and Duration Estimation*

Each attraction is assigned a realistic time duration based on typical visitor behavior. Longer attractions—such as museums or historical monuments—are scheduled earlier in the day, while short attractions are used to fill in flexible gaps.

3) *Activity Balancing*

The system balances heavy and light activities to avoid overloading any single day. For example, a demanding trekking activity is not paired with multiple physically intense experiences on the same day.

4) *Logical Morning–Evening Segmentation*

Attractions are placed in optimal time slots. Morning slots may include long-wait attractions or scenic sunrise views, while evenings may feature waterfronts, relaxing spots, or shopping.

5) *Breaks, Meals, and Buffer Time*

The methodology incorporates natural breaks and adequate buffer time to account for real-world delays, ensuring the itinerary is practical and traveler-friendly.

E. *Internal Consistency and Feasibility Checks*

After grouping and sequencing are completed, the system conducts internal checks to ensure:

- No attraction is scheduled outside its typical visiting hours.
- The travel route does not involve unnecessary backtracking.
- The total hours planned per day remain within reasonable limits.

F. *Structured Output Generation*

Finally, the itinerary is formatted into a clear, readable structure. Each day appears with

This ensures the output is not just functional but visually smooth and easy to navigate. Travelers can quickly interpret the flow and adjust plans if needed.

G. *Proposed Algorithm*

To ensure the system consistently produces valid and logically sequenced itineraries, we implemented a **Context-Aware Prompt Chaining Algorithm**. This algorithm standardizes the interaction between the user constraints and the Large Language Model (LLM), serving as a safeguard against common AI errors such as hallucinations or formatting inconsistencies.

The core logic, detailed in **Algorithm**, follows a "Generate-then-Validate" approach. Unlike simple chatbot interactions, this system strictly enforces input feasibility before invoking the costly AI inference step. Post-generation, it employs a schema validator to ensure the output can be parsed by the frontend application.

Algorithm : Intelligent Itinerary Generation Strategy

Input: User Request Vector (U) containing: Dest: Destination Days: Duration of trip Budget: Economic constraints Interests: Set of user preferences

Output: Structured Itinerary Object (I_final) or Error State

Algorithm 1: Itinerary Generation Logic

Input:
User Preference Vector
U=Destination,Days,Budget,InterestsU = \Destination, Days, Budget, Interests\U=Destination,Days,Budget,Interests

Output:
Structured Itinerary Object IfinalI_ final Ifinal or Error State

Constants:
MAX_DAYS = 15 (token-limit constraint)
MAX_RETRIES = 3 (error-recovery limit)

Algorithm 1: Itinerary Generation Logic

Input: User data
Output: Generated itinerary

1. **Compute Input Feasibility:**
   IF (. < 1)OR (. >
   _)THEN
   Return *Error("Duration out of bounds")*.
   END IF
2. **Construct Context:**
   Role ← "Expert Travel Planner".
   Schema ← JSON_Structure_Rules.
   Prompt ← Combine(Role, U, Schema).
3. **Initialize:**
   Attempts ← 0.
4. while Attempts < MAX_RETRIES do
5. Response ← Call_GenAI_API(Prompt).
6. try
7. I ← Parse_JSON(Response).
8. IF Validate_Logic(I) THEN
9. Return .
10. END IF
11. catch Parsing_Error
12. Attempts ← Attempts + 1.
13. end try
14. end while
15. Return *Error("Generation Failed")*.

Data Formatting

Data formatting is a vital part of creating a usable itinerary. Without standardization, the resulting plan can appear cluttered, confusing, or overwhelming. To avoid this, the system maintains strict formatting rules ensuring consistency across days, activities, and descriptions.

One significant challenge with LLMs is hallucination or inconsistent formatting. To mitigate this, the system enforces a JSON schema validation on the output. If the LLM output fails validation, a regeneration request is triggered automatically.

H. Latency Optimization

As noted in user interactions, generation latency was initially high (~15s). By optimizing the prompt length and switching to a faster model variant, the latency was reduced to approximately 5 seconds, providing a near-real-time experience.

IV. Results and Discussion

The The performance of the proposed AI-based itinerary planner was evaluated using a hybrid methodology comprising both quantitative metric-based assessment and qualitative human-centric review. The system was tested against a dataset of 50 distinct travel queries ranging from simple weekend getaways to complex, multi-city international trips.

In many scenarios, users noted that the generated plans resembled those crafted by human tour planners. The system's consistency stood out, particularly in maintaining the sequence of days and preventing schedule conflicts. The adaptability across different destinations further supports the system's usefulness for broader travel applications.

Furthermore, scenario-based evaluations demonstrated that the system could handle varied demands—from short weekend getaways to extended multi-day vacations—with effective prioritization, sorting, and time allocation.

A. Quantitative Performance Analysis

To measure the system's efficiency, we compared it against two baselines: (1) Manual Planning (using standard search engines like Google/TripAdvisor) and (2) A standard Rule-Based Recommendation System.

- *Latency:* The system achieved an average end-to-end response time of **4.8 seconds** per itinerary.
- *Constraint Satisfaction Rate (CSR):* defined as the percentage of user constraints (Budget + Duration + Interest) successfully met in the final output. Our system achieved a **92% CSR**, with failures primarily occurring in "Ultra-Low Budget" scenarios where the model struggled to find realistic accommodation prices.

B. Qualitative Case Studies (Scenario Analysis)

We conducted a blind evaluation where human reviewers (N=20) rated generated itineraries on a scale of 1-5 for "Logistical Feasibility."

Scenario 1: 3-Day Cultural Trip to Jaipur (Budget: Low)

Constraint Handling: The model demonstrated "Zero-Shot Spatial Reasoning." It correctly grouped geographically proximate locations (e.g., *City Palace*, *Jantar Mantar*, and *Hawa Mahal*) into the same morning slot to minimize travel costs, recommending "E-Rickshaws" over taxis to adhere to the budget constraint.

Outcome: The itinerary was rated **4.8/5** for feasibility.

Scenario 2: 5-Day Leisure Trip to Kerala (Budget: Luxury)

Context Awareness: The system successfully inferred implicit requirements. Although "Spa" was not explicitly requested, the model correlated the "Luxury" and "Kerala" tags to suggest an Ayurvedic Wellness session in Kumarakom. It also recommended a premium houseboat stay, distinguishing it from standard budget options.

Outcome: The itinerary was rated **4.7/5**, with minor deductions for suggesting a 6-hour drive without a lunch break in one instance.

C. Sensitivity Analysis (Ablation Study)

We investigated the impact of the Temperature hyperparameter on the quality of output.

- *Low Temperature (T=0.2):* Resulted in highly deterministic but generic outputs. Every "Paris" itinerary looked identical, lacking serendipitous discovery.
- *High Temperature (T=1.0):* Increased creativity but introduced "Hallucinations." In one test, the model invented a "Floating Market" in a city where none existed.
- *Optimal Setting (T=0.7):* This value (selected for the final prototype) balanced variety with factual grounding, maintaining a **Hallucination Rate of <5%** across all tests.

D. Error Analysis and Limitations

Despite the high success rate, two primary classes of errors were identified:

1. *Temporal Hallucination:* The model occasionally suggested visiting attractions on days they are typically closed (e.g., recommending the *Taj Mahal* on a Friday). This is a limitation of the pre-trained nature of LLMs lacking real-time calendar access.
2. Pricing Drift: While the model understands relative budget tiers (Low vs. High), specific price estimates (e.g., "Entry fee: ₹500") were often outdated due to inflation occurring after the model's training cutoff.

Conclusion

This research presented a novel approach to automated travel planning by integrating Large Language Models (LLMs) with a structured application layer. The developed system successfully addresses the limitations of traditional recommendation engines—which often provide fragmented lists—by synthesizing cohesive, day-wise itineraries that respect user constraints such as budget, interest, and pace. By leveraging the semantic reasoning capabilities of Google's Gemini model and the speed of FastAPI, the system achieves a balance between creative personalization and logical feasibility.

The experimental results demonstrated a 92% Constraint Satisfaction Rate and a high degree of logistical accuracy, validating the efficacy of the prompt engineering strategies employed. The system effectively mitigates common LLM issues, such as hallucinations, through strict schema enforcement and context injection. It offers a scalable solution that reduces the cognitive load on travelers, transforming hours of manual research into a near-instantaneous, actionable plan.

However, the study also identified limitations regarding real-time data accuracy, specifically concerning dynamic pricing and variable opening hours. **Future work** will focus on bridging this gap by integrating real-time APIs (such as Google Maps Platform or Amadeus) to validate the LLM's suggestions against live data. Additionally, we aim to expand the system to support multi-modal outputs, providing users with map visualizations and image galleries alongside their text-based itineraries, further enhancing the user experience in the smart tourism ecosystem.

Acknowledgment

I would like to thank the faculty of the Dept. of Information Technology at MMMUT, Gorakhpur for providing the technical environment and support necessary to complete this research

References

[1] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, "Mobile recommender systems in tourism," *Journal of Network and Computer Applications*, vol. 39, pp. 319–333, 2014.

[2] U. Gretzel, "Intelligent systems in tourism: A social science perspective," *Annals of Tourism Research*, vol. 38, no. 3, pp. 757-779, 2011.

[3] S. Li, J. Cao, and V. C. S. Lee, "A Comprehensive Survey on Large Language Models for Recommendation," *arXiv preprint arXiv:2305.06999*, 2023.

[4] H. Liu, et al., "LLM-Rec: Personalized Recommendation via Prompting Large Language Models," *arXiv preprint arXiv:2307.15780*, 2023.

[5] M. Nilashi, et al., "Travelers decision making using online review in social network sites: A fuzzy-logic approach," *Tourism Management*, vol. 59, pp. 233-242, 2017.

[6] Google AI, "Gemini API Documentation," [Online]. Available: https://ai.google.dev/docs. [Accessed: Nov. 20, 2025].

[7] F. Ricci, "Travel Recommender Systems," in *IEEE Intelligent Systems*, vol. 17, no. 6, pp. 55-57, 2002.

[8] J. Borràs, A. Moreno, and A. Valls, "Intelligent tourism recommender systems: A survey," *Expert Systems with Applications*, vol. 41, no. 16, pp. 7370-7389, 2014.

[9] A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[10] D. Buhalis and R. Law, "Progress in information technology and tourism management: 20 years on and 10 years after the Internet—The state of eTourism research," *Tourism Management*, vol. 29, no. 4, pp. 609-623, 2008.

[11] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734-749, 2005.

[12] Z. Xiang and D. R. Fesenmaier, "Big data analytics, tourism design and smart tourism," *Tourism Geographies*, vol. 19, no. 1, pp. 107-124, 2017.

[13] H. Werthner and F. Ricci, "E-commerce and tourism," *Communications of the ACM*, vol. 47, no. 12, pp. 101-105, 2004.

[14] K. Kabassi, "Personalization systems for tourism: a comparative study," *User Modeling and User-Adapted Interaction*, vol. 20, no. 1, pp. 83-118, 2010.

[15] T. B. Brown et al., "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[16] B. Pan, T. MacLaurin, and J. C. Crotts, "Travel blogs and the implications for destination marketing," *Journal of Travel Research*, vol. 46, no. 1, pp. 35-45, 2007.

[17] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1-38, 2019.

[18] D. Wang, S. Park, and D. R. Fesenmaier, "The role of smartphones in mediating the touristic experience," *Journal of Travel Research*, vol. 51, no. 4, pp. 371-387, 2012.

[19] OpenAI, "GPT-4 Technical Report," *arXiv preprint arXiv:2303.08774*, 2023.

[20] H. Lin, Y. Fan, and P. J. Zhang, "A Deep Learning Approach for Next Point-of-Interest Prediction in Tourism," *IEEE Access*, vol. 8, pp. 18933-18943, 2020.