Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 17, No. 3, 2026

A Large Language Model-Driven Intelligent Route Planning Framework for Personalized Tourism Navigation in Scenic Areas

Kai Ma *, Wei Pan

Institute for Big Data Research, Liaoning University of International Business and Economics, Dalian, Liaoning, 116052, China

*** Corresponding author:** Kai Ma (Email: makai@luibe.edu.cn)

Abstract: A tourist may describe a desired day as 'relaxed, coastal, suitable for parents, and not too crowded,' whereas a route optimizer requires numerical attributes and explicit constraints. This paper connects these two representations without asking a large language model (LLM) to draw the route itself. The LLM parses a natural-language request into a preference profile; a semantic-spatial network then links that profile to attraction attributes, travel connections, visit durations, and congestion information. Route selection is performed by a multi-criteria model that evaluates preference fit together with distance and time costs. The framework is examined using six attractions in Dalian and four traveler profiles. Compared with the shortest-path baseline, the LLM-assisted method increases the reported preference-matching degree by about 29.1%, although it does not always return the minimum-distance itinerary. The result suggests a practical division of labor: language modeling handles ambiguous user intent, while an explicit optimizer remains responsible for spatial feasibility and resource limits.

Keywords: Large Language Model; Intelligent Tourism; Route Planning; Personalized Recommendation.

1. Introduction

Digital travel services now support tasks that were formerly handled through maps, guidebooks, and local enquiry. Their value is not limited to finding a location; users also expect help deciding which places fit the purpose, pace, and circumstances of a trip. This expectation makes itinerary construction different from ordinary point-to-point navigation [4,16,27].

A shortest-path engine answers a well-defined spatial question, but many tourism requests are not stated in spatial terms [5,6]. Two visitors starting at the same location may want markedly different days: one may value museums and historical interpretation, while a family group may favor accessible entertainment and short walking segments. If distance is the only objective, both users can receive nearly identical sequences even though neither has asked for the same experience [1–4].

LLMs offer a way to interpret these open-ended requests [25,26,29]. Their relevant capability here is not unrestricted text generation, but extraction of entities, constraints, and relative preferences from ordinary language. A phrase such as 'traveling with elderly parents' can be mapped to walking tolerance and route pace; 'avoid busy places' can modify a congestion penalty. This translation provides information that a conventional optimizer cannot infer from coordinates alone.

Language interpretation does not remove the need for route computation. An LLM may recommend attractive places that cannot be visited within the available time or that form an inefficient sequence [25,26]. A usable system must therefore separate the semantic question—what does the visitor value?—from the spatial question—what ordered set of visits is feasible?

The main technical gap lies at this boundary. Tourism chatbots and recommender systems can produce suggestions, while mathematical planners require structured parameters [16–24]. Natural-language preferences are often ambiguous, graded, or context dependent, so they cannot be inserted directly into an objective function [25,26,29]. Conversely, a planner that accepts only manually selected categories loses much of the information expressed in the original request [7,16,17].

We address this gap with a semantic-spatial pipeline. The LLM is restricted to producing a structured preference record. Attraction attributes and transport links are represented in a weighted tourism network, and a multi-objective model—not the LLM—selects the itinerary. This arrangement keeps the final route subject to time and accessibility constraints while retaining information from the user's wording.

2. Related Work

2.1. Traditional Tourism Route Planning Methods

Early route-planning formulations treat attractions as vertices and travel links as weighted edges [1–4]. Dijkstra's algorithm and A* provide efficient solutions when the objective is a least-cost path [5,6]. That formulation is appropriate for navigation between fixed endpoints, but a sightseeing day also involves choosing which vertices to visit and how long to remain at each one.

The Orienteering Problem (OP) captures this selection aspect by maximizing collected value under a resource budget [1]. Tourist Trip Design Problems (TTDPs) add visit duration, opening or time constraints, and tourism-specific utility [2,4]. Personalized variants further assign different values to attractions for different visitors [3,7]. These models establish the optimization basis used in the present work.

Their input requirements are nevertheless highly structured. Interest weights, visit limits, and attraction scores must normally be specified before search begins. A request written as a sentence must therefore be reduced manually to the parameters expected by the model. Our work retains the OP/TTDP view of route feasibility but introduces an automated interpretation stage ahead of optimization.

2.2. Multi-objective Optimization for Tourism Planning

Tourism itineraries expose competing objectives. Reducing travel distance can leave more time for sightseeing, but it may exclude an attraction that closely matches the visitor's interests. Avoiding congestion may require a detour, and adding another stop can increase preference satisfaction while making the day infeasible. A single scalar such as distance does not describe these trade-offs. [8–14]

Evolutionary search is often used when several objectives and combinatorial constraints must be handled together [8]. NSGA-II, for example, maintains non-dominated alternatives through sorting and elitist selection [9], while ant-colony and particle-swarm methods provide other global-search strategies [10,11]. Gaspar-Cunha and Covas discuss the broader use of evolutionary algorithms for multi-objective problems [8].

Tourist-guide and team-orienteering studies show how time windows, resource limits, and destination value can be combined operationally [12,13]. More recent recommender research also considers satisfaction together with system-level criteria such as accuracy and fairness [14]. These studies demonstrate that competing criteria can be optimized, but they generally assume that preference variables already exist.

Accordingly, the unresolved issue for this paper is upstream of the optimizer. A multi-objective method can balance supplied weights; it cannot determine what a traveler meant by 'a quiet cultural afternoon.' We use the LLM to infer that structured input and leave the trade-off calculation explicit.

2.3. AI-based Tourism Recommendation

Tourism recommenders estimate which destinations a user is likely to value. Collaborative filtering derives this estimate from behavior shared across users, content-based systems compare user and item attributes, and hybrid systems combine both sources [16]. Their output is commonly a ranked list rather than a timed and connected itinerary.

Content-based methods are useful when attractions have descriptive attributes [17]. Location-based social-network research extends the representation with geographic and social signals, allowing point-of-interest rankings to reflect both interest and place [18,19]. Location, however, is usually one feature in ranking and not a guarantee that the complete sequence is feasible.

Deep and hybrid models have been used to learn latent tourism preferences, including signals extracted from reviews and observed trajectories [20–23]. Social-media content provides another source of travel intent and destination perception [24]. These approaches can improve prediction when sufficient historical data are available.

Dependence on past clicks, ratings, or trajectories creates difficulty for a new user and for a request that differs from previous behavior. Moreover, selecting individually suitable places does not specify their order, accumulated travel time, or accessibility. The proposed framework therefore treats recommendation scores as inputs to a constrained route problem rather than as the final answer.

2.4. Large Language Models in Tourism

Large language models changed the interface to recommendation by accepting unconstrained text and producing context-sensitive responses [25,26,29]. Few-shot language modeling and instruction following explain why a model can extract a requested schema from examples and directions [25,26]. For tourism, this makes it possible to collect preferences conversationally instead of presenting a long fixed questionnaire.

Studies of artificial intelligence in travel and hospitality emphasize both service opportunities and changes in tourist interaction [27,28]. An LLM can combine statements about companions, pace, attraction type, and aversion to crowds within one exchange [25,26,29]. This flexibility is especially relevant when the user has no historical profile.

Free-form generation also introduces risks. Suggested attractions may be outdated or fictitious, and an itinerary stated fluently may violate travel-time or connectivity constraints. General LLM surveys similarly distinguish language-generation capability from external factual grounding [29]. For route planning, the model output should therefore be bounded by a known attraction dataset and checked by a deterministic optimization layer.

Our design adopts this constrained role. The LLM returns preference fields and weights but cannot add an attraction to the map or declare an infeasible edge valid. This differs from using a chatbot as the route generator and creates an auditable connection between the visitor's words and the numerical route objective.

3. Methodology

3.1. Framework Overview

The framework separates interpretation, environmental representation, optimization, and presentation. This separation matters because each stage has a different correctness criterion: semantic extraction should reflect the request, the tourism model should contain valid places and links, the optimizer should satisfy constraints, and the interface should explain the selected route.

Fig 1 shows four connected layers. The intention layer produces a structured traveler profile. The knowledge layer supplies attraction attributes, spatial links, and context. The optimization layer searches over feasible visit sequences using both sets of information. The recommendation layer returns the selected itinerary and accepts revisions. Data move forward through the pipeline, while user feedback can initiate a new interpretation and search cycle.

At the first layer, the input remains the traveler's own wording. The parser looks for attraction themes, available time, preferred pace, walking tolerance, group composition, and sensitivity to congestion. Missing fields are left unspecified or handled by system defaults; they are not silently inferred as fixed facts about the traveler.

The semantic-spatial layer is the grounding source. Every candidate attraction has coordinates, category labels, an expected visit duration, and other attributes. Edges record whether two places can be connected and at what travel cost. Consequently, a preference score can raise an attraction's value without bypassing the network that determines whether it can be visited.

The route layer evaluates candidate sequences rather than isolated attractions. Its objective rewards semantic fit and useful sightseeing time, while penalizing distance and congestion. Hard limits on total time, attraction count, and connectivity remove candidates that cannot be implemented.

The final layer presents the sequence and its principal attributes. If a user changes the available time or rejects a crowded stop, the altered request is parsed again and the route is recalculated. The interaction is thus a controlled re-optimization, not an unconstrained continuation of generated advice.

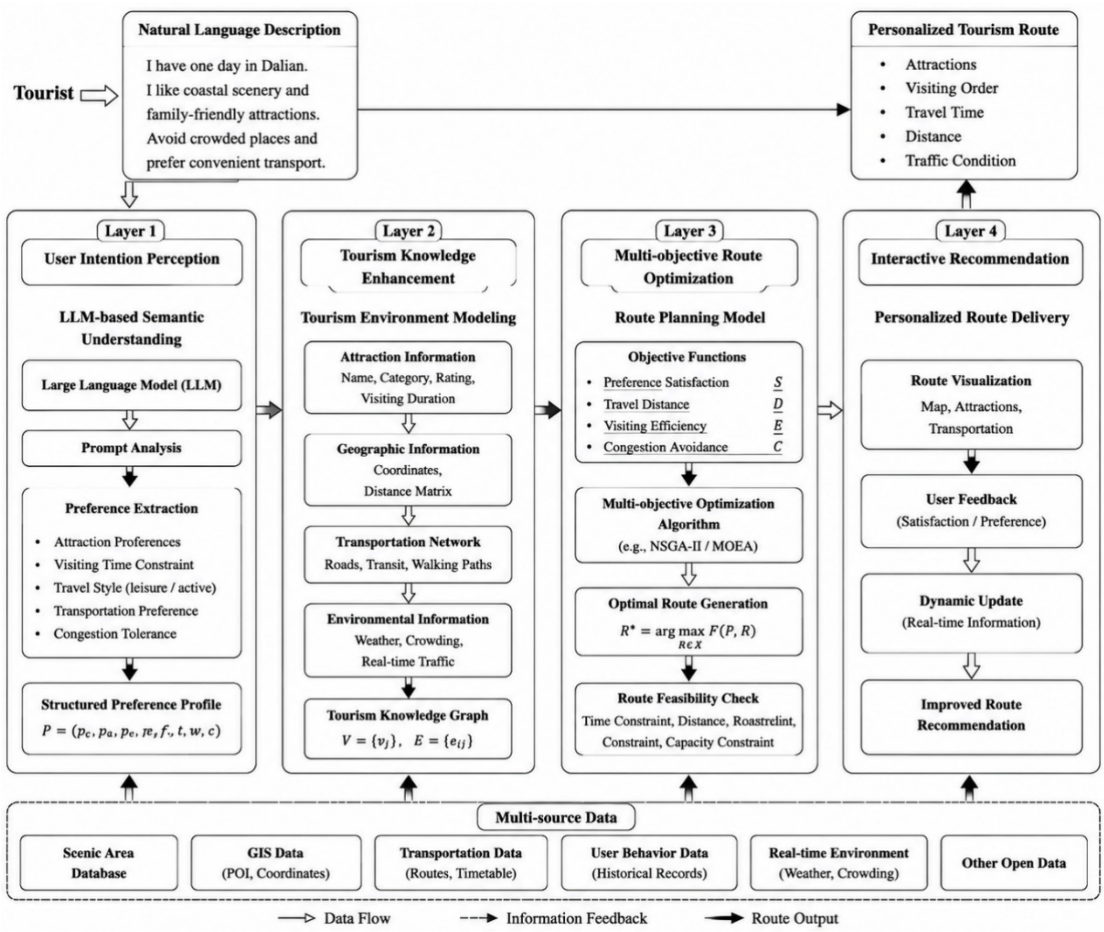

Fig 1. Overall architecture of the proposed LLM-driven intelligent tourism route planning framework

Operationally, a request is converted to a profile, matched against grounded attraction attributes, and supplied to the multi-objective solver. The solver returns an ordered feasible set, after which the interface can describe why the chosen places fit the stated interests.

3.2. LLM-based User Requirement Understanding and Preference Extraction

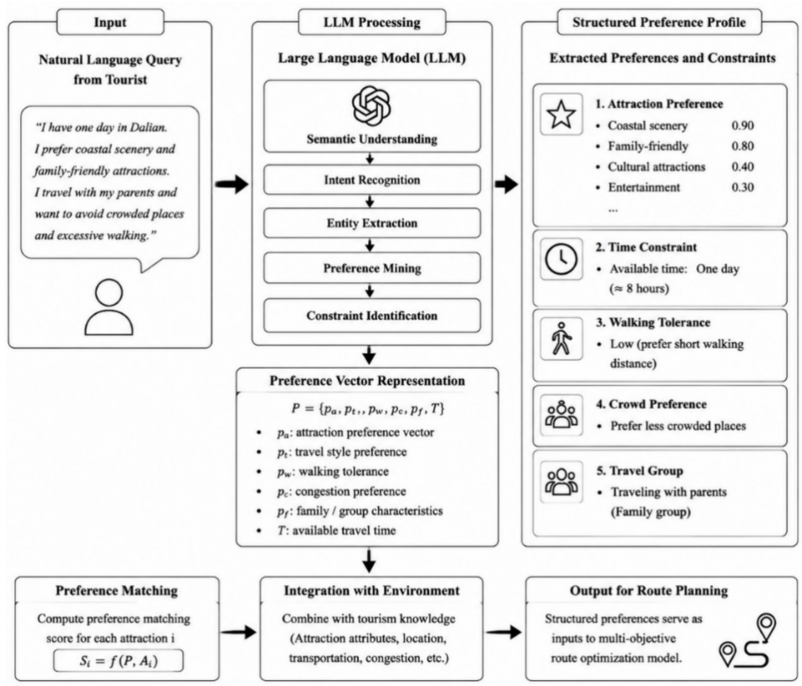

Fig 2. Workflow of LLM-based user requirement understanding and preference extraction

Tourism requests mix explicit constraints with softer language. 'One day' gives a time budget, while 'not too tiring' expresses a graded preference that may depend on age or group composition. Fixed checkboxes capture the first type easily but often lose the nuance of the second.

We use the LLM as a schema-filling component. A prompt defines the allowed fields and requests values for attraction category, duration, pace, walking tolerance, crowd preference, and travel party. The returned record is validated before it is accepted by the optimizer.

The extraction sequence in Fig 2 has three steps: receive the original request, interpret its entities and constraints, and serialize the result as a preference profile. Keeping these steps visible supports inspection when an extracted value does not agree with the user's wording.

Semantic interpretation first identifies time expressions, companions, desired activities, and avoidance statements. It then maps them to the allowed vocabulary and assigns relative scores where the request indicates strength. The result is a compact record, rather than a prose itinerary.

The mapping from the request to that record is denoted by:

$$\mathrm{P} = \mathrm{LLM}(\mathrm{X}) \tag{1}$$

Here, the input variable is the traveler's natural-language description and the output variable is the validated preference profile. For the example request concerning one day in Dalian, coastal scenery, family-friendly places, elderly parents, low crowding, and limited walking, the output contains six corresponding dimensions:XP

$$\mathrm{P} = \{\mathrm{p_a}, \mathrm{p_t}, \mathrm{p_w}, \mathrm{p_c}, \mathrm{p_f}, \mathrm{T}\} \tag{2}$$

The components represent attraction interest, preferred travel pace, walking tolerance, crowd tolerance, party characteristics, and available time, respectively.$\mathrm{p_a p_t p_w p_c p_f T}$

Attraction interest may contain several weighted themes rather than one category. Pace controls how aggressively visits are scheduled; walking and crowd tolerance alter relevant penalties; party characteristics help distinguish, for example, family-accessible facilities; and available time is applied as a hard budget.

To connect the profile with a candidate attraction i, both are embedded or encoded in a common attribute space and compared through the matching function:

$$\mathrm{S_i} = \mathrm{f}(\mathrm{P}, \mathrm{A_i}) \tag{3}$$

The attraction vector describes candidate i, and the resulting score measures its compatibility with the traveler profile. The score is bounded and stored with the node for route evaluation.$\mathrm{A_i S_i}$

This procedure gives the LLM a narrowly defined responsibility. It interprets language, but the knowledge network determines what exists and the optimization model determines what can be scheduled. The preference score is therefore evidence used by the planner, not a route asserted by the model.

3.3. Tourism Semantic-Spatial Knowledge Model Construction

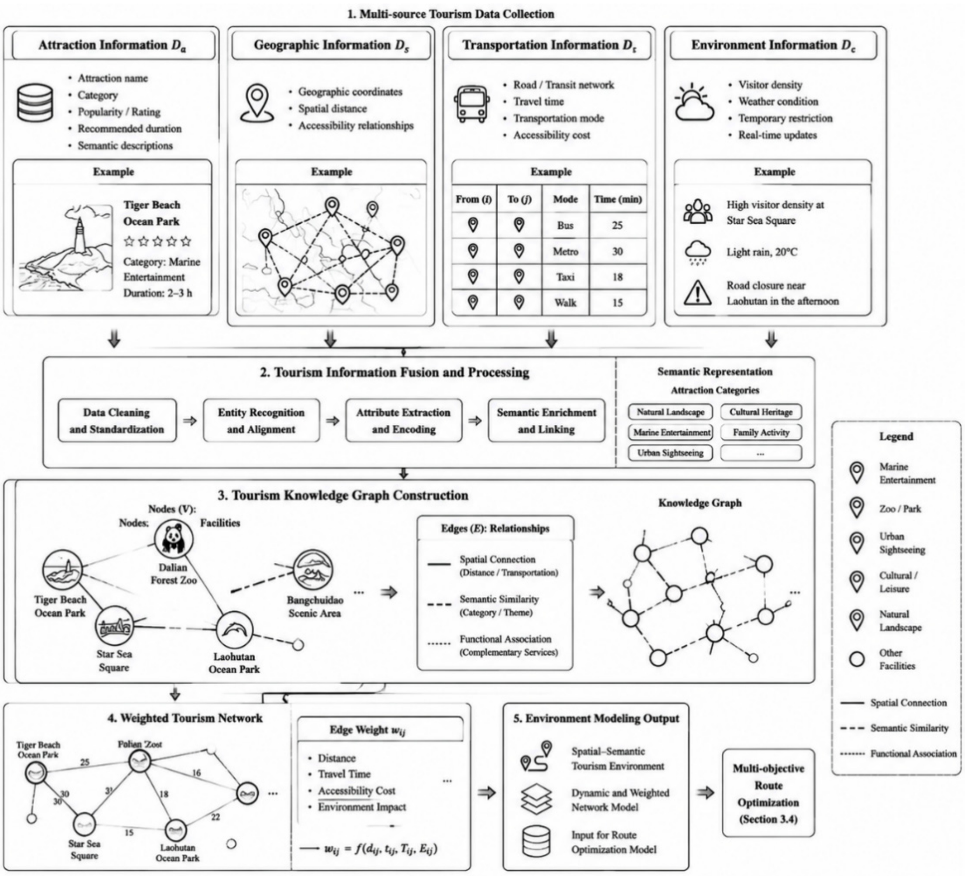

Fig 3. Construction process of tourism knowledge graph and environment modeling

Preference scores are meaningful only when attached to a grounded representation of the destination. We build a semantic-spatial network that joins descriptive attraction attributes with coordinates, visit times, transport connections, and contextual penalties. Ratings alone are insufficient because two highly rated places may be impractical to combine in a limited day.

Fig 3 outlines the construction process. Source records are cleaned and reconciled, attraction descriptions are mapped to the common semantic vocabulary, spatial and transport relationships are computed, and the result is stored as a weighted network. The same identifiers are used by both the matching and routing stages.

The destination model is expressed as the weighted graph:

$$G = (V, E, W) \tag{4}$$

The node set contains attractions and relevant service facilities, the edge set contains valid travel connections, and the weight set stores node and edge attributes. A node is represented by:$VEWv_i \in V$

$$v_i = (x_i, y_i, c_i, r_i, t_i, s_i) \tag{5}$$

Its fields include geographic coordinates, attraction category, popularity or rating, recommended visit duration, and semantic descriptor.$x_i y_i c_i r_i t_i s_i$

Semantic descriptors use the vocabulary needed by preference extraction, including natural landscape, cultural heritage, marine activity, family activity, and urban sightseeing. Shared labels allow the profile and attraction to be compared without relying on attraction names alone.s_i

3.3.1. Multi-source Tourism Information Integration

The integrated source collection is written as:

$$D = \{D_a, D_g, D_t, D_e\} \tag{6}$$

Its terms denote attraction records, geographic data, transportation data, and context that may change over time.$D_a D_g D_t D_e$

Attraction records provide names, categories, ratings, visit durations, and descriptions. Coordinates and accessibility data establish spatial relations. After unit and identifier normalization, the geographic distance between attractions i and j is computed as:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{7}$$

where the distance variable refers to the pair of attraction nodes i and j.$d_{ij} ij$

This geometric value is then amended by the available transport connection rather than used as the complete edge cost.

$$C_{ij} = d_{ij} + \lambda T_{ij} \tag{8}$$

The travel-cost term combines separation and transport time with the stated weighting coefficient. Context such as crowd level, weather, or a temporary closure contributes a further adjustment:$C_{ij} T_{ij} \lambda$

$$E_{ij} = C_{ij}(1 + \gamma R_{ij}) \tag{9}$$

The environmental term records that context and the coefficient controls its influence. In the present case study, these variables are treated as supplied attributes rather than continuously streamed observations.$R_{ij}\gamma$

3.3.2. Knowledge Graph Construction

Graph construction creates one node per grounded attraction or service facility. Edges are added only for accessible pairs, so semantic similarity cannot connect two places that lack a usable travel relation. This is the main safeguard against a plausible but spatially invalid LLM suggestion.

Node attributes hold both factual and semantic information. Relational edges may encode geographic connection, transport accessibility, and similarity of tourism function. For example, two marine attractions may be semantically close but spatially distant; the representation retains both facts instead of collapsing them into one score.

Weights translate the representation into a form usable by search. Preference matching affects node benefit, while distance, travel time, and context affect transition or visit cost. The optimizer can therefore compare routes that fit the traveler differently while applying the same feasibility rules.

3.3.3. Integration with Preference Extraction Module

For every candidate attraction, the profile from Section 3.2 is compared with its stored attributes using:PA_i

$$M_i = f(P, A_i) \tag{10}$$

The output is the matching degree between traveler vector P and attraction vector A. These node-level scores are passed to the objective in Section 3.4, completing the connection from the original request to route search.$M_i PA_i i$

3.4. LLM-guided Multi-objective Route Optimization Model

Route construction is posed as selection and ordering under a time budget. The objective does not assume that the most preferred set is automatically feasible: it evaluates semantic benefit together with travel, visit, and congestion costs for each candidate sequence.

An itinerary is an ordered list of visited attractions:

$$R = \{v_1, v_2, \dots, v_n\} \tag{11}$$

The route variable at position i denotes the i-th visit, and n is the number of included attractions.$v_i n$

The composite objective is:

$$\max F(R) = w_1 F_{pre}(R) + w_2 F_{time}(R) - w_3 F_{dis}(R) - w_4 F_{con}(R) \tag{12}$$

Its four quantities represent preference satisfaction, useful visit efficiency, travel-distance cost, and congestion cost. The associated weights expose the trade-off and can be adjusted for different operating policies. $F_{pre}(R) F_{time}(R) F_{time}(R) F_{con}(R) w_i$

Preference benefit is derived from the profile-attraction comparison, not from the LLM's prose output. For attraction i, the matching calculation is:

$$S_i = f(P, A_i) \tag{13}$$

P is the validated traveler profile and A_i is the stored attribute vector. Route-level preference satisfaction is then accumulated as:

$$F_{pre}(R) = (\Sigma S_i) / n \tag{14}$$

A larger value means that the selected set aligns more closely with the expressed interests. The spatial penalty uses the sum of distances between successive visits:

$$F_{dis}(R) = \Sigma d_{i,\, i+1} \tag{15}$$

Here, d between two consecutive nodes is obtained from the tourism network. Total duration adds both dwell and transfer time:

$$T(R) = \Sigma t_i + \Sigma \tau_{i,\, i+1} \tag{16}$$

The visit-time term belongs to attraction i, while the transfer-time term belongs to the edge leading to the next attraction.

The efficiency component relates achieved sightseeing benefit to the limited time resource:

$$F_{time}(R) = (\Sigma S_i) / T(R) \tag{17}$$

This term discourages spending an excessive share of the day in transfer. Crowding is modeled separately because a short route through heavily congested places may still be undesirable:

$$F_{con}(R) = \Sigma C_i \tag{18}$$

C_i is the crowd value associated with attraction i; increasing it raises the route penalty.

Three hard constraints remove impractical candidates. First, accumulated visit and transfer time cannot exceed the traveler-provided budget:

$$T(R) \leq T_{max} \tag{19}$$

The right-hand side is the maximum available duration. Second, the visit count is bounded:T_{max}

$$n_{min} \leq n \leq n_{max} \tag{20}$$

The lower and upper limits prevent an empty itinerary and an unrealistically dense schedule.$n_{min}n_{max}$

Third, every consecutive pair must correspond to a valid network edge:

$$e_{ij} \in E \tag{21}$$

E is the set of accessible connections. A high semantic score cannot override this condition.

The selected route is the feasible sequence with the best weighted score. In this formulation, language-derived preference affects utility, whereas the network and constraints retain control of spatial and temporal validity.

4. Case Study

4.1. Study Area and Dataset Construction

Dalian provides a compact test setting with coastal, ecological, cultural, and urban attractions. The city was chosen because contrasting attraction types make traveler preferences observable in the selected route. The case study evaluates the framework's computational behavior on this constructed dataset; it is not presented as a complete operational tourism platform.

Six sites are included: Xinghai Square, Bangchuidao Scenic Area, Tiger Beach Ocean Park, Dalian Forest Zoo, Jinshitan National Tourism Resort, and Donggang Business District. Together they cover urban sightseeing, natural scenery, marine entertainment, family activity, and coastal recreation. This variation allows the same origin area to yield different itineraries for different profiles.

For each site, the dataset records coordinates, attraction category, descriptive labels, recommended visit time, and accessibility to other sites. Nodes correspond to sites and weighted edges to usable transfers. Descriptions are normalized to the vocabulary used by the preference extractor.

Candidate routes can consequently be evaluated on both sides of the problem: how well their nodes match the profile and how costly their ordered connections are. Congestion and time variables provide the remaining case-study inputs.

4.2. Experimental Scenarios

Three planners are evaluated under the same attraction network.

(1) Shortest-path baseline (SP)

SP minimizes accumulated geographic distance:

$$\min F_1 = D \tag{22}$$

The objective supplies a distance-efficient reference but contains neither profile matching nor visit utility.D

(2) Multi-objective planner with manual preferences (MO)

MO uses manually assigned attraction-satisfaction weights in the objective:

$$F = w_1 S - w_2 D - w_3 T \tag{23}$$

The terms are attraction satisfaction, distance, and total time, with their respective coefficients. Because the user must specify the preference parameters directly, this baseline isolates the value of semantic extraction.$SDTw_i$

(3) LLM-assisted multi-objective planner (LLM-MO)

LLM-MO performs the following operations:

(1) receive the natural-language request;

(2) extract and validate the traveler profile;

(3) match the profile to grounded attraction attributes;

(4) solve the constrained multi-objective route problem; and

(5) return the ordered itinerary with an explanation of the match.

The illustrative request specifies one day in Dalian, interest in coastal scenery and marine culture, travel with parents, a relaxed pace, and limited walking. It is converted to the structured parameter record:

$$U = \{I, T, W, C\} \tag{24}$$

The fields represent interest, available time, walking tolerance, and crowd tolerance. Only these validated values enter the optimizer.ITWC

All methods use the same attraction data and environmental settings. Repeated runs are averaged where the search method is stochastic. The comparison records route distance, use of the time budget, preference matching, and the combined satisfaction score.

4.3. Experimental Scenarios

Four traveler profiles are used to vary attraction theme, pace, time allowance, and physical or crowd-related preference. The scenarios are designed to test whether the method returns different feasible sequences from the same attraction network rather than a single generic route.

Table 1. Experimental Scenarios, Tourist Requirements, and Main Optimization Objectives

Scenario	Tourist Requirement	Main Optimization Objectives
Cultural and Urban Exploration	"I want to explore the urban characteristics of Dalian and visit representative city landmarks. The travel duration is one day."	urban landscape preference; cultural experience; moderate travel distance.
Family-oriented Tourism	"I travel with children and prefer attractions suitable for family activities, marine entertainment, and educational experiences."	family suitability; entertainment value; convenient transportation.
Coastal Leisure Tourism	"I prefer natural scenery and coastal landscapes. I hope to enjoy a relaxing seaside trip."	natural landscape similarity; scenic experience; low travel intensity.
Short-time Efficient Travel	"I only have four hours before departure. Please provide the most efficient sightseeing route."	time efficiency; attraction importance; transportation convenience.

4.4. Evaluation Indicators

Evaluation uses four reported measures.

(1) Route distance (RD)

RD sums the length of all consecutive transfers:

$$RD = \sum_{i=1}^{n-1} d(i, i+1) \tag{25}$$

Lower RD indicates less travel between selected attractions, but does not measure semantic fit.

(2) Time-utilization efficiency (TUE)

TUE is the share of available time used for attraction visits:

$$TUE = \frac{T_v}{T_a} \tag{26}$$

The numerator is total sightseeing time and the denominator is the visitor's time budget.T_vT_a

(3) Preference-matching degree (PMD)

PMD aggregates semantic agreement across selected attractions:

$$\mathrm{PMD} = \frac{\sum_{i=1}^{n} p_i T_i}{\sum_{i=1}^{n} T_i} \tag{27}$$

Each term combines profile-attraction similarity with the attraction-importance weight.$p_i T_i$

(4) Overall satisfaction score (OSS)

OSS combines the preceding quality dimensions:

$$\mathrm{OSS} = \alpha\mathrm{PMD} + \beta\mathrm{TUE} - \gamma\mathrm{RD} \tag{28}$$

The coefficients determine the contribution of each component to the reported score.

4.5. Results and Discussion

4.5.1. Route Planning Results

SP selects the geographically compact sequence, as expected from its objective. Its route is useful as a distance lower reference, yet it cannot distinguish two travelers who share the same start and time budget but express different interests.

MO changes the selected attractions when manually assigned preferences change. This confirms that multi-objective search can represent a personalized trade-off, while also exposing its input burden: the user or system designer must translate the request into weights before planning.

LLM-MO performs that translation from the original text and then uses the same grounded search logic. Differences from MO should therefore be interpreted as the effect of extracted profile values, not as an LLM generating an unconstrained sequence.

The selected examples make this distinction visible. For the family profile, Tiger Beach Ocean Park and Dalian Forest Zoo receive higher matching scores and long walking transfers are penalized. For coastal leisure, Bangchuidao and Jinshitan gain value through landscape and seaside attributes. The route changes because node utilities change, while accessibility and time rules remain fixed.

4.5.2. Quantitative Comparison

LLM-MO reports the strongest combined performance. Its PMD is approximately 29.1% higher than that of SP, quantifying the loss incurred when distance alone represents route quality. PMD also exceeds the manually parameterized MO result, indicating that the profile extracted from the request carries useful differentiation in these scenarios. This comparison does not show that language interpretation improves the geometric optimum; rather, it shows that a modest spatial trade-off can purchase a closer match to the stated visit purpose.

Table 2. reports the numerical comparison for the three planners

Method	RD (km)	TUE (%)	PMD (%)	OSS
SP	18.6	71.4	62.5	68.3
MO	20.1	78.6	79.2	80.5
LLM-MO	19.3	85.7	91.6	89.8

5. Discussion

The experiment supports a division between semantic interpretation and route optimization. SP is efficient in distance, MO handles explicit trade-offs, and LLM-MO reduces the effort needed to express those trade-offs numerically. The principal gain is therefore at the human-model interface, while feasibility continues to come from the network and constraints.

Scenario-specific routes also show why a single destination ranking is insufficient. Changes in companions, pace, and walking tolerance alter not only which attractions are desirable but also which combinations fit within the same day. Evaluating ordered sequences captures this interaction.

The architecture can accept additional attributes such as live crowd estimates, weather, and transport status because these variables already have defined places in the graph and objective. Doing so would require reliable data feeds and a policy for update frequency; an LLM alone cannot supply current operational values.

Several limitations qualify the results. The attraction set is small, environmental inputs are largely static, and the study does not report field observations of traveler behavior. Preference extraction may also be sensitive to prompt design and model version. Future work should test larger networks, repeated user interactions, uncertain or contradictory requests, live crowd and transport data, and robustness across different LLMs.

6. Conclusion

This paper presented a grounded route-planning pipeline in which an LLM converts a travel request into a structured profile and a separate multi-objective solver constructs the itinerary. On the Dalian case, the method increases preference matching by about 29.1% over shortest-path planning while preserving explicit time and accessibility constraints. The finding is limited to the reported dataset, but it illustrates a useful design principle for smart-tourism systems: natural-language models should interpret the traveler, whereas verified spatial data and optimization rules should determine the route.

Acknowledgments

This work was supported by the “Research Project of Liaoning University of International Business and Economics” [2023XJLXYBWT04].

References

[1] Vansteenwegen, P., et al. (2011). The orienteering problem: A survey. European Journal of Operational Research, 209(1), 1–10. https://doi.org/10.1016/j.ejor.2010.03.045.

[2] Gavalas, D., Konstantopoulos, C., Mastakas, K., et al. (2014). A survey on algorithmic approaches for solving tourist trip design problems. Journal of Heuristics, 20(3), 291–298. https://doi.org/10.1007/s10732-014-9242-5.

[3] García, A., Arbelaitz, O., Linaza, M. T., et al. (2010). Personalized tourist route generation. In Lecture Notes in Computer Science. Springer. https://doi.org/10.1007/978-3-642-16985-4_47.

[4] Vansteenwegen, P., & Souffriau, W. (2011). Tourist trip planning functionalities: State–of–the–art and future. Information Technology & Tourism, 12(4), 305–315. https:// doi. org/10.3727/109830511X13049763021853.

[5] Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. Numerische Mathematik, 1(1), 269–271. https:// doi. org/10.1007/BF01386390.

[6] Hart, P. E., Nilsson, N. J., & Raphael, B. (1972). A formal basis for the heuristic determination of minimum cost paths. IEEE Transactions on Systems Science & Cybernetics, 4(2), 100–107. https://doi.org/10.1109/TSSC.1968.300118.

[7] Lim, K. H., Chan, J., Leckie, C., et al. (2015). Personalized tour recommendation based on user interests and points of interest

visit durations. In Proceedings of the AAAI Conference on Artificial Intelligence. AAAI Press.

[8] Gaspar-Cunha, A., & Covas, J. A. (2008). Multi-objective optimization using evolutionary algorithms. Computational Optimization and Applications, 39(1), 75–96. https://doi.org/ 10. 1007/s10589-007-9053-9.

[9] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation, 6(2), 182–197. https://doi.org/10.1109/4235.996017.

[10] Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation, 1(1), 53–66. https://doi.org/10.1109/4235.585892.

[11] Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. Swarm Intelligence, 1(1), 33–57. https://doi.org/ 10. 1007/s11721-007-0002-0.

[12] Vansteenwegen, P., & Van Oudheusden, D. (2007). The mobile tourist guide: An OR opportunity. OR Insight, 20(3), 21–27. https://doi.org/10.1057/ori.2007.17.

[13] Souffriau, W., Vansteenwegen, P., Van den Berghe, G., et al. (2013). The multiconstraint team orienteering problem with multiple time windows. Transportation Science, 47(1), 53–63. https://doi.org/10.1287/trsc.1120.0416.

[14] Zaizi, F. E., Qassimi, S., & Rakrak, S. (2025). Multi-objective optimization for personalized and fairness recommender systems: A hybrid approach. SN Computer Science, 6(6). https://doi.org/10.1007/s42979-025-04230-8.

[15] Yu, Z. X. Learning travel recommendation from user-generated GPS trajectories [Manuscript].

[16] Ricci, F., Rokach, L., & Shapira, B. (Eds.). (2011). Recommender systems handbook. Springer-Verlag New York. https:// doi.org/10.1007/978-0-387-85820-3.

[17] Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In F. Ricci, L. Rokach, & B. Shapira (Eds.), Recommender systems handbook (pp. 73–105). Springer. https://doi.org/10.1007/978-0-387-85820-3_3.

[18] Gao, H., Tang, J., Hu, X., & Liu, H. (2015). Content-aware point of interest recommendation on location-based social networks. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. AAAI Press.

[19] Zhang, J. D., & Chow, C. Y. (2013). iGSLR: Personalized geo-social location recommendation. In Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM. https://doi.org/ 10. 1145/ 2525314.2525339.

[20] Fathima, I., & Kotaiah, B. (2022). Deep learning based tourism recommendation system. SSRN Electronic Journal. https:// doi. org/ 10.2139/ssrn.4286575.

[21] Z., J. H., & Li, M. (2019). Research on recommendation system based on tourism user data and comments. Computer Engineering & Software.

[22] Chen, D., Ong, C. S., & Xie, L. (2016). Learning points and routes to recommend trajectories. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. ACM. https://doi.org/ 10.1145/ 2983323. 2983672.

[23] Aravindharaj, T., Rohit, T., Raj, P., et al. (2017). Recommendation systems using hybrid collaborative filtering.

[24] Xiang, Z., & Gretzel, U. (2010). Role of social media in online travel information search. Tourism Management, 31(2), 179–188. https://doi.org/10.1016/j.tourman.2009.02.016.

[25] Brown, T. B., Mann, B., Ryder, N., et al. (2020). Language models are few-shot learners [Preprint]. arXiv. https://doi. org/ 10. 48550/arXiv.2005.14165.

[26] Ouyang, L., Wu, J., Jiang, X., et al. (2022). Training language models to follow instructions with human feedback [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2203.02155.

[27] Gidumal, J. B. (2020). Impact of artificial intelligence in travel, tourism, and hospitality. In Encyclopedia of Tourism Management and Marketing. Springer. https://doi.org/ 10. 1007/ 978-3-030-05324-6_110-1.

[28] Tussyadiah, I. (2020). A review of research into automation in tourism: Launching the Annals of Tourism Research Curated Collection on Artificial Intelligence and Robotics in Tourism. Annals of Tourism Research, 81. https://doi.org/ 10. 1016/j. annals. 2020.102883.

[29] Zhao, W., Ma, Y., Wen, J., et al. (2026). A survey of large language models. Frontiers of Computer Science, 20(12). https:// doi.org/10.1007/s11704-026-60308-3.