Decoupling Preference Aggregation and Interest Evolution via Multi-Agent Reasoning for Next POI Recommendation

WeiWen Cao
Department of Computer Science
Graduate School of Systems and Information Engineering
University of Tsukuba
Tsukuba, Japan
s2420682@u.tsukuba.ac.jp

Takashi Inui
Department of Computer Science
Graduate School of Systems and Information Engineering
University of Tsukuba
Tsukuba, Japan
inui@cs.tsukuba.ac.jp

***Abstract*—Point-of-interest (POI) recommendation aims to predict a user's next visit based on historical mobility behaviors and is fundamental to location-based services. While large language models (LLMs) have shown promising reasoning capabilities for this task, existing LLM-based approaches still face several challenges: (i) they often rely on monolithic reasoning processes, making it difficult to explicitly aggregate heterogeneous user preference signals; (ii) they lack effective mechanisms to capture the temporal evolution of user interests, resulting in limited modeling of dynamic mobility intentions; and (iii) they struggle to efficiently filter large-scale POI candidates, which increases computational cost and degrades recommendation precision.**

In this paper, we proposed a *Preference Aggregation and Evolution-aware Multi-Agent* framework for next POI recommendation (PAE-MA). PAE-MA explicitly decouples preference aggregation from evolution-aware reasoning and decomposes the recommendation process into three collaborative stages: multi-granularity preference aggregation, collaborative candidate filtering, and spatio-temporal evolution-aware reranking. The framework is purely train-free and compatible with off-the-shelf LLMs, without requiring task-specific fine-tuning. Extensive experiments demonstrate the effectiveness of the proposed approach across multiple evaluation settings.

***Index Terms*—Point-of-interest recommendation, large language models, multi-agent systems.**

I. Introduction

Location-based social networks (LBSNs) produce vast mobility data, enabling Point-of-Interest (POI) recommendation to predict users' next visits for applications like travel planning and marketing [1], [2]. Recently, Large Language Models (LLMs) have emerged as powerful tools for this task, leveraging their prior knowledge, common-sense mobility patterns, and reasoning capabilities [3].

Existing LLM-based approaches often rely on Supervised Fine-Tuning (SFT) for accuracy [4]–[6], yet the intensive data and computational demands of SFT can diminish the models' inherent reasoning flexibility.In contrast, prompt-based methods bypass expensive training to preserve LLMs' natural reasoning for interactive recommendation [7], [8], with recent trends employing structured agent workflows and tool augmentation to further enhance performance [9], [10]

Despite these advances, existing LLM-based approaches face several challenges: **(i) Implicit preference entanglement.** Loose reasoning structures often entangle heterogeneous signals [11], failing to separate stable long-term preferences from short-term intentions, which limits controllability. **(ii) Static interest modeling.** LLMs struggle with sequential patterns and temporal dependencies [12], often treating preferences as static and failing to capture interest evolution under changing contexts. **(iii) Inefficient candidate filtering.** Reasoning over an overly broad search space increases computational costs and degrades recommendation precision. Consequently, existing methods fail to capture dynamic interest evolution under shifting contexts. Furthermore, inefficient candidate filtering remains a challenge, as reasoning over an excessive search space increases computational overhead and compromises recommendation precision.

To address these limitations, we proposed a fully train-free **Preference Aggregation and Evolution-aware Multi-Agent** framework (**PAE-MA**) for POI recommendation. PAE-MA decouples preference aggregation from evolution-aware reasoning through three collaborative modules: **(i) Weighted preference aggregation** summarizes stable preferences from historical behaviors across multiple granularities. **(ii) Collaborative candidate filtering** employs multiple agents to narrow down candidates into a focused set. **(iii) Spatio-temporal evolution-aware reranking** models dynamic interest evolution to reorder POIs based on context. This structured paradigm enables interpretable, prompt-driven decision-making compatible with off-the-shelf LLMs.

Our contributions can be summarized as follows:

- We proposed **PAE-MA**, the first train-free multi-agent framework that decouples preference aggregation from interest evolution for structured, interpretable reasoning.
- We designed an aggregation–evolution decoupled reasoning paradigm that decomposes POI recommendation into three collaborative modules, including **weighted aggregation**, **collaborative filtering**, and **evolution-aware reranking**, enabling fine-grained modeling of stable pref-

erences and dynamic interest evolution within a unified framework.
- Extensive experimental results demonstrate the effectiveness of PAE-MA framework across multiple evaluation settings.

II. Related Work

A. Next POI Recommendation

Next POI recommendation was initially framed as a sequential prediction task using personalized Markov chains [13], [14], though these struggled with complex behaviors. Deep learning shifted the focus to RNNs for capturing spatio-temporal dynamics [15]–[17], followed by attention and Transformer architectures to model long-range dependencies and periodic patterns [18]–[21], For example, iPCM [22] introduces personalized spatio-temporal clustering to constrain the candidate space for next POI recommendation within a Transformer-based architecture.

Graph-based methods further enhanced this by encoding higher-order collaborative signals via GNNs [23]–[26], GMCA [27] incorporates multiple context-aware signals to jointly model spatial relationships and user check-in behaviors. Additionally, contrastive learning has been employed to improve representation robustness through multi-view spatio-temporal augmentation [28]–[32]. However, these traditional approaches rely on task-specific training and lack interpretability, limiting their flexibility.

B. LLM-Based Recommender Systems

LLMs have redefined recommendation as generative language tasks, leveraging their reasoning and contextual prowess [33]–[35]. Beyond domain-specific tasks, multimodal recommendation has also been widely studied [36]–[40]. More recently, agent-based frameworks have also been explored in related predictive tasks [41]. In POI recommendation, LLMs model sequential behaviors through semantic representations and prompt-based reasoning [5], [7], [8]. For example, Tool4POI [10] augments LLMs with external tools for preference-aware retrieval and reranking in next POI recommendation. Recent trends have shifted toward preference alignment [42], [43], grounding user behaviors in semantic identifiers consistent with human preferences.

Despite their potential, most existing LLM methods encode preferences implicitly, lacking fine-grained control and explicit modeling of preference evolution for robust zero-shot generalization.

III. Methodology

As illustrated in Figure 1, We propose PAE-MA, which decomposes the inference process into three functional stages. Each stage assigns the LLM a specific role, enabling task-aligned reasoning throughout the next-POI recommendation pipeline. Instead of directly predicting the next POI from raw check-in sequences, our framework progressively infers user preferences, constructs candidate POIs, and performs spatio-temporal reasoning.

A. Problem Definition

Let U and V denote the sets of users and POIs. For a user $u \in U$, their historical behavior is represented by a chronological sequence $H_u = \{v_1, v_2, \ldots, v_n\}$, where $v_i \in V$. This is augmented with timestamps to form a time-aware sequence H_u^t for generating user preference evolution T_u.

The goal is to generate a ranked list of POIs by leveraging an LLM to infer structured preferences S_u^{pref} and evolution patterns T_u. To facilitate knowledge access and candidate construction, the LLM utilizes a unified toolset $\mathcal{D}_{tool}$:

$$\mathcal{D}_{tool} = \{\mathcal{D}_{search}, \mathcal{D}_{retrieval}\}, \tag{1}$$

where $\mathcal{D}_{\text{search}}$ provides POI metadata and $\mathcal{D}_{\text{retrieval}}$ enables preference-aware filtering. Given instruction I and initial context C_{init}, the LLM-based agent generates the final recommendation list $\mathcal{C}_u$ as:

$$\mathcal{C}_u = \text{LLM}(S_u^{pref}, T_u, C_{\text{init}}, \mathcal{D}_{\text{tool}}). \tag{2}$$

B. Long-Short-Term User Preference Aggregation

To capture latent intentions from sparse POI sequences, we propose a module that transforms raw records into structured semantic representations via three stages: (i) sequence construction, (ii) LLM-based preference summarization, and (iii) structured aggregation.

1) User Preference Construction: We decompose the historical sequence H_u into long-term habits (H_u^{long}) and short-term intentions (H_u^{short}) to preserve their distinct roles. Given training sequences H_u^{train} (length l) and test interactions H_u^{test} (length s), the components are defined as:

$$H_u^{long} = \begin{cases} (v_{l-99}, \ldots, v_l), & l \geq 100 \\ H_u^{\text{train}}, & \text{otherwise} \end{cases} \tag{3}$$

$$H_u^{short} = \begin{cases} (v_{s-N_{\text{short}}+1}, \ldots, v_s), & s \geq N_{\text{short}} \\ H_u^{\text{test}}, & \text{otherwise} \end{cases} \tag{4}$$

The unified representation $H_u = H_u^{long} \cup H_u^{short}$ integrates stable patterns with recent signals, providing a comprehensive foundation for subsequent semantic preference modeling.

2) LLM-based Preference Summarization: To enrich the semantically sparse H_u, we leverage an LLM to analyze behaviors augmented by external POI knowledge. We define a contextual input $I_{context}$ as:

$$I_{context} = \{\textit{Prompt}_{pref}, H_u, \mathcal{D}_{search}\} \tag{5}$$

where $\mathcal{D}_{search}$ interfaces with a POI database $\mathcal{DB}_{poi}$. The LLM invokes this tool to retrieve metadata K_{meta} (including categories and Google Plus Code regions):

$$K_{meta} = \text{LLM}_{search}(H_u \mid \mathcal{DB}_{poi}) \tag{6}$$

Based on $I_{context}$ and K_{meta}, the LLM generates a natural-language preference summary T_u^{pref}:

$$T_u^{pref} = \text{LLM}_{summarize}(I_{context}, K_{meta}) \tag{7}$$

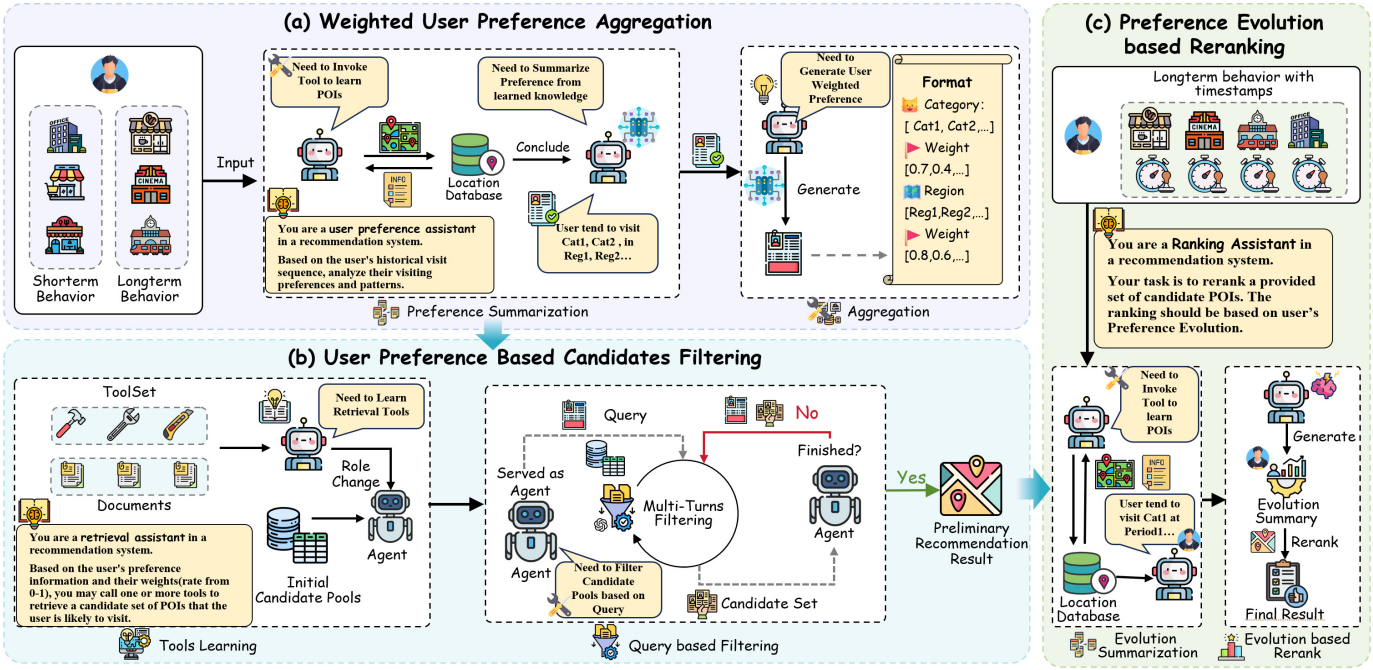

Fig. 1: **Overall framework of PAE-MA.**

3) Structured Preference Aggregation: To enable precise filtering, the unstructured T_u^{pref} is mapped into structured, executable signals S_u^{pref} via an LLM aggregator:

$$S_u^{pref} = \text{LLM}_{aggr}(T_u^{pref}) \tag{8}$$

S_u^{pref} encodes actionable constraints like category distributions and spatial transitions, facilitating efficient retrieval in subsequent stages.

Preference Aggregation Prompt

You are a recommendation assistant. Analyze the user's visit sequence to identify preferences across *region* and *category* dimensions. **[Output Format]**

- **regions**: [List likely regions]
- **region_focus**: [Attention weights (0-1) for each region]
- **categories**: [List likely POI categories]
- **category_focus**: [Attention weights (0-1) for each category]

C. Multi-Agents Collaborative Candidate Filtering

To efficiently filter the large-scale POI set $\mathcal{V}$ while preserving semantic and spatial consistency, we define multiple functional roles that perform specialized filtering tasks, coordinated by an arbitration mechanism to prevent over-filtering.

1) Agent Role Definition: We define three functional agents: **CategoryAgent** enforces category-level constraints from S_u^{pref}; **RegionAgent** ensures spatial coherence relative to the current location v_s; and **ArbitratorAgent** resolves conflicts and controls filtering intensity.

2) Transition-based Candidate Initialization: To alleviate cold-start effects, we initialize the search space using historical transition patterns. Given the last visited POI v_s, the initial set $\mathcal{C}_{init}$ is:

$$\mathcal{C}_{init} = \{v_j \in \mathcal{V} \mid P(v_j \mid v_s) > 0\} \tag{9}$$

3) Collaborative Filtering Process: At iteration t, the CategoryAgent and RegionAgent sequentially refine the candidates:

$$\mathcal{C}_u^{(t,\text{cat})} = \mathcal{D}_{\text{retrieval}}\big(\mathcal{C}_u^{(t-1)} \mid S_u^{pref}\big) \tag{10}$$

$$\mathcal{C}_u^{(t,\text{reg})} = \mathcal{D}_{\text{retrieval}}\big(\mathcal{C}_u^{(t,\text{cat})} \mid S_u^{pref}\big) \tag{11}$$

The ArbitratorAgent then applies an arbitration rule based on a decision function $\text{Accept}(\cdot)$ to balance strictness and diversity:

$$C_u^{(t)} = \begin{cases} C_u^{(t,\text{reg})} \cap C_u^{(t-1)}, & \text{if Accept}(C_u^{(t,\text{reg})} \mid S_u^{pref}) \\ C_u^{(t-1)}, & \text{otherwise} \end{cases} \tag{12}$$

4) Ranking and Selection: Upon termination, candidates are ranked by spatial proximity to v_s:

$$\mathcal{C}_u^{rank} = \text{DistanceSort}\big(\mathcal{C}_u^{(t)} \mid v_s\big) \tag{13}$$

Finally, the top-K candidates are selected for the subsequent reasoning stage:

$$\tilde{\mathcal{C}}_u = \text{Top} - \text{K}\big(\mathcal{C}_u^{rank}\big) \tag{14}$$

D. Spatio-Temporal Evolution-aware Reranking

Aggregated preferences capture stable interests but overlook dynamic evolution. To model the coupled dynamics of temporal context and spatial transitions, we introduce a re-ranking stage where an LLM acts as a spatio-temporal reasoning agent to adjust rankings based on recent trends.

1) Evolution Context and Summary Generation: Let N_{evo} be the window length. We construct an evolution-aware representation H_u^t from the N_{evo} most recent behaviors, incorporating POIs (v_i), temporal segments (τ_i), and intervals (Δt_i):

$$H_u^t = \left\{ \left(v_{l-N_{\text{evo}}+1}, \tau_{l-N_{\text{evo}}+1}, \Delta t_{l-N_{\text{evo}}+1}\right), \ldots, \left(v_l, \tau_l, \Delta t_l\right) \right\} \tag{15}$$

Guided by $\textit{Prompt}_{\textit{evolution}}$, the LLM analyzes H_u^t to generate an evolution summary T_u that characterizes emerging interest shifts and transition tendencies:

$$T_u = \text{LLM}_{\textit{summarize}}\left(\textit{Prompt}_{\textit{evolution}}, H_u^t, \mathcal{D}_{\textit{search}}\right) \tag{16}$$

2) Evolution-guided Reranking: Finally, the agent reranks the preliminary candidate list $\tilde{\mathcal{C}}_u$ from the filtering stage by aligning it with the inferred evolution trends in T_u:

$$\mathcal{C}_u = \text{LLM}_{\textit{rank}}\left(\tilde{\mathcal{C}}_u \mid T_u\right) \tag{17}$$

This joint consideration of preference consistency, spatial plausibility, and interest evolution ensures the final output $\mathcal{C}_u$ prioritizes POIs that reflect the user's immediate trajectory and latent intent.

Evolution Extraction Prompt

You are a user behavior analyst. Analyze the following POI interaction sequence to identify behavioral evolution and shifts.

[Task]

- Cluster history by temporal proximity and feature similarity.
- Detect evolving patterns and significant shifts in behavior over time.

[Output Format]

- **Analysis**: [Brief analysis]
- **Behavior Patterns**: [Pattern description]
- **Regular Temporal Clustering?**: [yes/no/other]

IV. Experiments

In this section, we conduct a comprehensive experimental evaluation of the proposed method from multiple perspectives. Our analysis is organized around the following research questions:

- **RQ1 (Overall Analysis):** How does our approach perform compared to baseline methods?
- **RQ2 (Backbone Stability):** How does the proposed method generalize across different LLM backbones?
- **RQ3 (Ablation Study):** How does ablation study validate the effectiveness of key components in our approach?
- **RQ4 (Hyperparameter Analysis):** How do different hyperparameters affect the model performance?
- **RQ5 (Efficiency and Cost-Effectiveness):** Does the proposed method achieve a balance between recommendation performance and computational cost?
- **RQ6 (Case Study):** How does the proposed method leverage preference aggregation and evolution-aware reranking to generate interpretable next-POI recommendations?

A. Experimental Setup

a) Datasets and Preprocessing.: We evaluate our method on three real-world datasets: Foursquare-NYC, Foursquare-TKY [44], and Gowalla-CA [45]. Following standard protocols [25], we filter users and POIs with fewer than 10 interactions and sort check-ins chronologically. For each user, the last visited POI serves as the ground truth, with prior interactions used for history modeling. Table I summarizes the dataset statistics.

b) Parameter Settings.: To capture stable and dynamic interests, we set sequence lengths for long-term preference (H_u^{long}), short-term preference (N_{short}), and evolution modeling (N_{evo}) to 100, 30, and 100, respectively. These values remain fixed across all experiments. After multi-round filtering, the top $K = 20$ candidates are retained for the final reranking.

c) Implementation Details.: We use GPT-3.5-turbo as the backbone LLM for preference aggregation, filtering, and reranking. To ensure generation stability, we set both the temperature and top-p value to 1.0 across all stages.

TABLE I: Statistics of three real-world datasets

Dataset	Users	POIs	Check-ins
NYC	1,083	5,135	147,938
TKY	2,293	7,873	447,570
CA	6,592	14,027	351,857

d) Evaluation Metrics.: We evaluate task performance using Acc@K ($K \in \{1, 5, 10\}$) and MRR@10. Acc@K measures the hit ratio of the ground-truth POI within the top-K recommendations. MRR@10 assesses the ranking quality by calculating the mean reciprocal rank:

$$\text{MRR@}K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{r_u} \cdot \mathbb{I}(r_u \leq K) \tag{18}$$

where $|\mathcal{U}|$ is the number of test users, r_u is the rank of the target POI for user u, and $\mathbb{I}(\cdot)$ is the indicator function. Together, these metrics reflect both retrieval accuracy and ranking effectiveness.

e) Baselines: We compare the proposed method with a diverse set of representative next-location recommendation approaches, including both traditional models and recent LLM-based methods.

- **Traditional baselines:** FPMC [46], PRME [47], STGCN [17], and GETNext [24], which model users' historical mobility patterns through sequential modeling or graph-based structures.
- **LLM-based baselines:** LLMMob [7], LMMove [8], ZeroPOIRec [48], LLMRank [12], and Tool4POI [10], which methods leverage LLMs to capture user preferences and perform POI recommendation.

B. Overall Analysis (RQ1)

Table II compares our method against various baselines. Key observations include:

TABLE II: Overall performance comparison on three real-world datasets. **Bold** indicates our result. Underline denotes the best traditional baseline. $\star$ denotes the best LLM-based baseline.

Model	NYC				TKY				CA			
	Acc@1	Acc@5	Acc@10	MRR@10	Acc@1	Acc@5	Acc@10	MRR@10	Acc@1	Acc@5	Acc@10	MRR@10
FPMC	0.1003	0.2126	0.2970	0.1701	0.0814	0.2045	0.2746	0.1344	0.0383	0.0702	0.1159	0.0911
LSTM	0.1305	0.2719	0.3283	0.1857	0.1335	0.2728	0.3277	0.1834	0.1321	0.2746	0.3233	0.1831
PRME	0.1159	0.2236	0.3105	0.1712	0.1052	0.2278	0.2944	0.1786	0.0521	0.1034	0.1425	0.1002
STGCN	0.1799	0.3425	0.4279	0.2788	0.1716	0.3453	0.3927	0.2504	0.0810	0.1842	0.2579	0.1675
GETNext	0.2435	0.5089	0.6143	0.3621	0.2254	0.4417	0.5287	0.3262	0.1526	0.3278	0.3946	0.2364
LLMRank	0.1247	0.3311	0.4034	0.2387	0.1586	0.3486	0.4013	0.2328	0.1667	0.3730	0.4008	0.2463
ZeroPOIRec	0.1474	0.4082	0.4668	0.2690	0.1936	0.4535	0.5132	0.2796	0.1718	0.4607	0.5224	0.2833
LMMove	0.1525	0.4429	0.5080	0.2931	0.1293	0.3601	0.4154	0.2248	0.1331	0.3831	0.4627	0.2462
LLMMob	0.2343	0.5066	0.5660	0.3502	0.1545	0.3837	0.4451	0.2519	0.1236	0.3694	0.4028	0.2078
Tool4POI	0.3174*	0.6643*	0.8333*	0.4722*	0.1962*	0.4905*	0.6265*	0.3300*	0.2510*	0.6076*	0.7768*	0.3994*
Ours	**0.3470**	**0.7078**	**0.8550**	**0.5054**	**0.2389**	**0.5428**	**0.6495**	**0.3697**	**0.2879**	**0.6313**	**0.7876**	**0.4332**
Improv.	***+7.0%***↑	***+6.6%***↑	***+2.6%***↑	***+7.0%***↑	***+6.0%***↑	***+10.7%***↑	***+3.7%***↑	***+12.0%***↑	***+13.6%***↑	***+3.9%***↑	***+1.4%***↑	***+8.5%***↑

a) Overall Performance.: Our method consistently achieves state-of-the-art results across all datasets and metrics. On NYC, it attains 0.3470 Acc@1 and 0.5054 MRR@10, surpassing the best baseline, Tool4POI. Similar gains on TKY and CA demonstrate its robustness across diverse urban scenarios.

b) Comparison with Traditional Methods.: Our approach significantly outperforms traditional sequence- and graph-based models (e.g., FPMC, PRME, STGCN, GETNext). For instance, our Acc@10 on NYC reaches 0.8550, far exceeding GETNext's 0.6143, indicating that our preference representation captures complex mobility patterns more effectively than conventional methods.

c) Comparison with LLM-based Baselines.: Compared to recent LLM-based models (e.g., LLMMob, LMMove, ZeroPOIRec, LLMRank, Tool4POI), our method shows consistent improvements. This highlights that while LLMs provide strong priors, explicit preference modeling and structured ranking objectives are essential for superior POI recommendation.

C. Stability Across LLM Backbones (RQ2)

To verify model-agnostic effectiveness, we evaluate PAE-MA against the competitive Tool4POI baseline across various LLMs, including `gpt-4o-mini`, `qwen-plus`, `claude-3.5-sonnet`, and `deepseek-v3.2`. As shown in Table III, our method consistently outperforms the baseline across all metrics and scales. For instance, it improves `gpt-3.5-turbo`'s Acc@1 from 0.317 to 0.347 and maintains a stable margin on the frontier `claude-3.5-sonnet`. These results confirm that our performance gains stem from architectural innovation rather than reliance on specific model knowledge, demonstrating the framework's robust versatility.

D. Ablation Study (RQ3)

Ablation experiments on Foursquare-NYC (Figure 2) evaluate the contribution of each key component:

- **w/o Long.** Removing long-term preferences causes a clear performance drop, particularly in Acc@10, confirming their necessity for capturing stable interests and global relevance.

TABLE III: Performance Comparison on NYC dataset across different backbones.

Backbone	**Method**	**Acc@1**	**Acc@5**	**Acc@10**	**MRR@10**
gpt-3.5-turbo	Baseline	0.317	0.664	0.833	0.472
	Ours	**0.347**↑	**0.708**↑	**0.855**↑	**0.505**↑
gpt-4o-mini	Baseline	0.280	0.659	0.821	0.439
	Ours	**0.317**↑	**0.688**↑	**0.826**↑	**0.478**↑
qwen-plus	Baseline	0.261	0.615	0.816	0.413
	Ours	**0.297**↑	**0.663**↑	**0.832**↑	**0.458**↑
claude-3.5-sonnet	Baseline	0.335	0.694	0.825	0.484
	Ours	**0.348**↑	**0.701**↑	**0.832**↑	**0.498**↑
deepseek-v3.2	Baseline	0.321	0.664	0.828	0.472
	Ours	**0.337**↑	**0.694**↑	**0.834**↑	**0.492**↑

- **w/o Short.** Excluding short-term preferences primarily degrades precision-oriented metrics like Acc@1, indicating their crucial role in identifying immediate user intents.
- **w/o Aggr.** Without explicit aggregation, the model relies on simplified constraints rather than weighted representations. The resulting performance decline proves aggregation's necessity for effective retrieval and reranking.
- **w/o Evo.** Removing the evolution-based reranking module leads to a noticeable decline in ranking quality (e.g., MRR@10), showing that static representations cannot replace dynamic interest evolution modeling.

Overall, these results confirm that long- and short-term signals are complementary, while preference aggregation and evolution-aware reranking jointly ensure recommendation accuracy and robustness.

E. Parameter Analysis(RQ4)

We conduct a series of experiments to investigate the impact of key hyperparameters on model performance. The results are illustrated in Fig. 3 and summarized as follows:

- **Short-term behavior length N_{short}.** The model achieves the best or near-optimal performance when $N_{\text{short}} = 20$ on both Acc@10 and MRR@10. Increasing the short-term window leads to slight performance degradation,

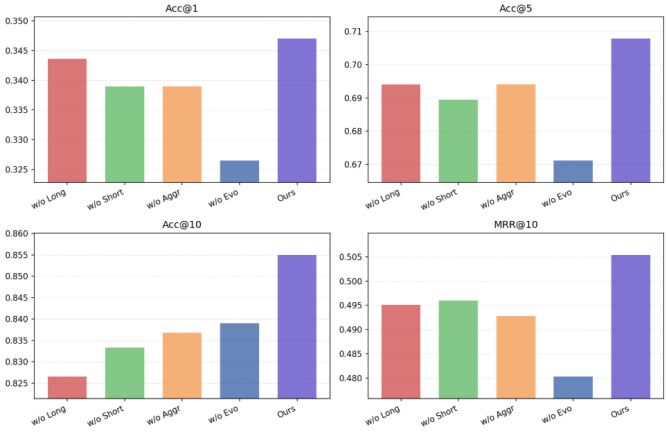

Fig. 2: Ablation study result on NYC dataset.

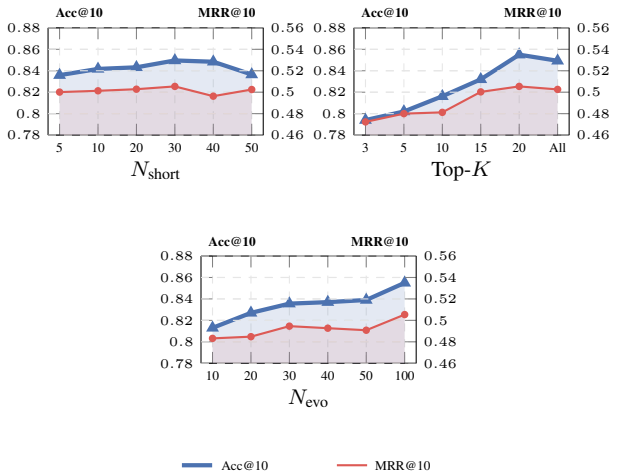

Fig. 3: Hyperparameter analysis on NYC dataset.

indicating that excessively long short-term sequences may introduce redundant or noisy signals.

- **Filtering threshold Top-K.** The overall performance peaks when Top-K is set to 20. A small candidate set limits the coverage of relevant POIs, while an overly large candidate set introduces many low-relevance candidates, increasing the difficulty of reranking.
- **Evolution window length N_{evo}.** Model performance shows an overall upward trend as N_{evo} increases and reaches its best performance at $N_{\text{evo}} = 100$. This suggests that a longer evolution window helps capture users' preference evolution more effectively.

F. Efficiency and Cost-Effectiveness Analysis (RQ5)

Table IV and Figure 4 compare the inference latency and token consumption of the proposed method against the baseline.

a) Module-wise Latency Reduction.: Our modular design significantly enhances efficiency. The optimized preference aggregation reduces *Preference* stage latency by 14.4% (NYC) and 30.3% (TKY). Crucially, evolution-aware pruning mitigates decision complexity, slashing *Rerank* latency by over 84.6% and 86.0%, respectively, thereby accelerating the final decision-making process.

b) Scalability and Robustness.: The efficiency gains scale positively with dataset size. On the larger TKY dataset, the latency reduction in retrieval and reranking stages substantially offsets the overhead of the evolution module, yielding a 12.8% decrease in total inference time. This confirms the framework's robustness for large-scale recommendation.

c) Cost-Effectiveness.: While evolution modeling marginally increases total token usage, it optimizes cost-effectiveness by reducing *Completion Tokens*—the most expensive part of LLM inference—by 1.3% (NYC) and 8.9% (TKY). As illustrated in Figure 4, despite the evolution module's token share (44.3% on NYC, 35.5% on TKY), it successfully compresses the reranking stage's token burden to below 11%, resulting in superior overall execution efficiency.

TABLE IV: Inference performance comparison focusing on latency and compressed token consumption.

Data	Model	Preference(s)	Filtering(s)	Evolution(s)	Rerank(s)	Total Time(s)	Comp. Token
NYC	Baseline	5716.3	5012.2	–	7338.7	18067.2	1,052,440
	Ours	4891.2	4886.3	7838.8	1130.3	18746.7	1,038,793
	Improv.	↓14.4%	↓2.5%	–	↓84.6%	↑3.8%	↓1.3%
TKY	Baseline	14205.1	22884.4	–	19327.1	56416.7	3,781,668
	Ours	9905.2	19191.8	17384.4	2701.9	49183.2	3,446,257
	Improv.	↓30.3%	↓16.1%	–	↓86.0%	↓12.8%	↓8.9%

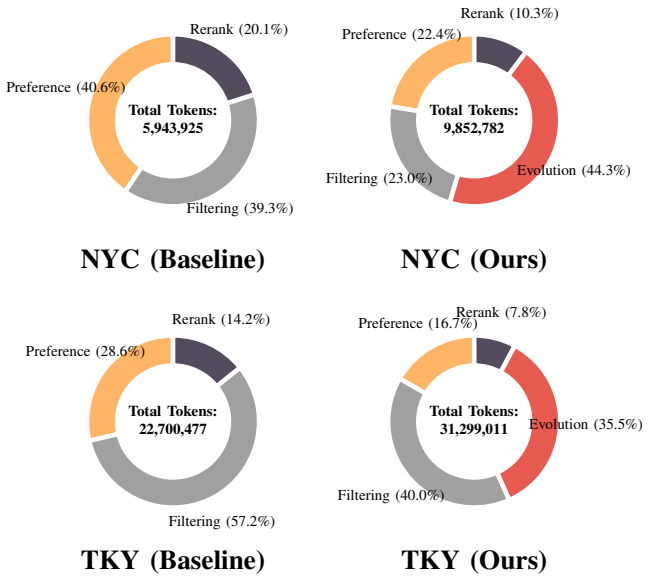

Fig. 4: Token consumption distribution (inner values represent total tokens).

G. Case Study (RQ6)

Figure 5 illustrates our framework's reasoning process. The Preference Agent first aggregates long- and short-term interests into structured semantic preferences to guide candidate filtering. Subsequently, the Evolution-aware Agent models temporal patterns and recent behavioral shifts to refine the ranking. By prioritizing POIs consistent with evolving interests, the framework successfully ranks the ground-truth POI at the top,

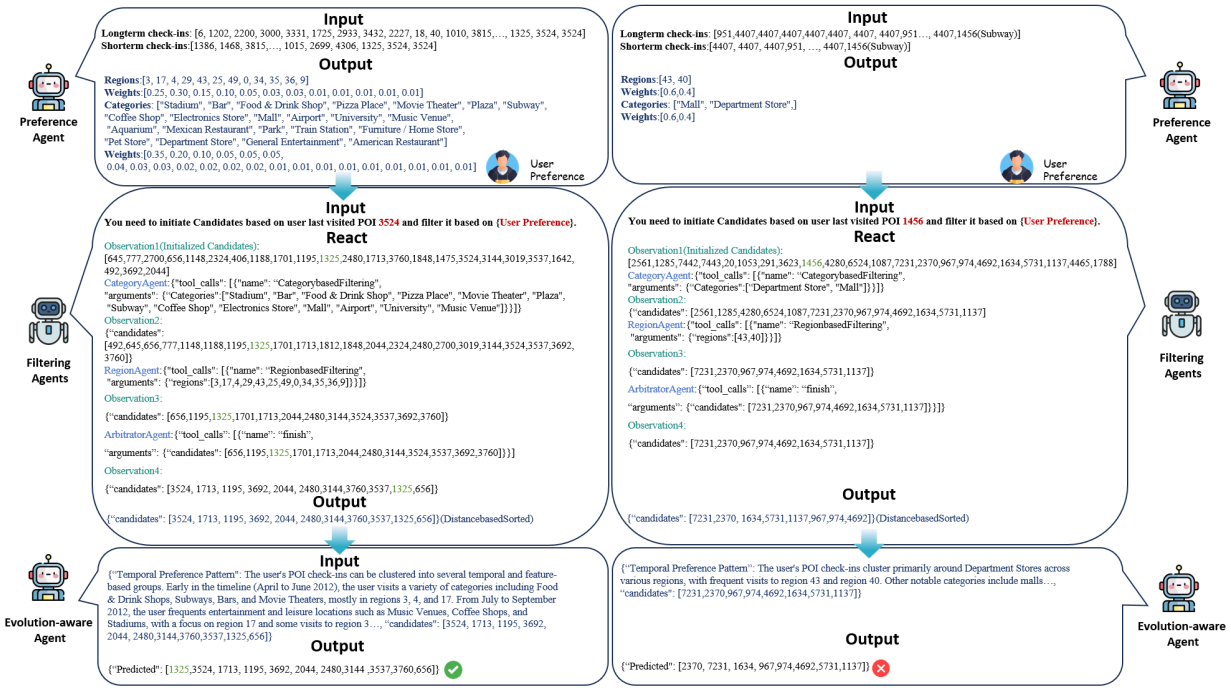

Fig. 5: A case study for PAE-MA. The ground-truth target POI is highlighted in green.

demonstrating the synergy between preference aggregation and evolution modeling. In contrast, the failure case in Figure 5 corresponds to highly repetitive user interactions, where the target POI is observed only once. Such sparse signals are easily dominated during candidate filtering and provide limited evidence for modeling interest evolution.

V. Conclusion

This paper presents PAE-MA, a multi-agent framework for next POI recommendation that decouples preference aggregation from interest evolution. Unlike monolithic LLM approaches, PAE-MA employs a structured, interpretable reasoning process across three stages: user weighted preference aggregation, candidate filtering, and evolution-aware reranking. By leveraging specialized agents, the framework efficiently handles large candidate spaces while capturing both stable preferences and dynamic interest shifts. Extensive experiments across multiple datasets validate PAE-MA's effectiveness and robustness, suggesting that explicit aggregation-evolution modeling is a promising direction for advancing LLM-based recommendation systems.

References

[1] Tianao Sun, Ke Fu, Weiming Huang, Kai Zhao, Yongshun Gong, and Meng Chen. Going where, by whom, and at what time: Next location prediction considering user preference and temporal regularity. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2784–2793, 2024.

[2] Zeyu Zhou, Yan Lin, Haomin Wen, Qisen Xu, Shengnan Guo, Jilin Hu, Youfang Lin, and Huaiyu Wan. Trajcogn: Leveraging LLMs for cognizing movement patterns and travel purposes from trajectories. *arXiv preprint arXiv:2405.12459*, 2024.

[3] Qianru Zhang, Peng Yang, Junliang Yu, Haixin Wang, Xingwei He, Siu-Ming Yiu, and Hongzhi Yin. A survey on point-of-interest recommendation: Models, architectures, and security. *IEEE Transactions on Knowledge and Data Engineering*, 2025.

[4] Shuai Liu, Ning Cao, Yile Chen, Yue Jiang, and Gao Cong. nextlocllm: next location prediction using LLMs. *arXiv preprint arXiv:2410.09129*, 2024.

[5] Peibo Li, Maarten de Rijke, Hao Xue, Shuang Ao, Yang Song, and Flora D. Salim. Large language models for next point-of-interest recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2024, page 1463–1472. ACM, July 2024.

[6] Wilson Wongso, Hao Xue, and Flora Salim. Genup: Generative user profilers as in-context learners for next poi recommender systems. In *Proceedings of the 33rd ACM International Conference on Advances in Geographic Information Systems*, pages 436–439, 2025.

[7] Xinglei Wang, Meng Fang, Zichao Zeng, and Tao Cheng. Where would i go next? large language models as human mobility predictors. *arXiv preprint arXiv:2308.15197*, 2023.

[8] Shanshan Feng, Haoming Lyu, Fan Li, Zhu Sun, and Caishun Chen. Where to move next: Zero-shot generalization of LLMs for next poi recommendation. In *2024 ieee conference on artificial intelligence (cai)*, pages 1530–1535. IEEE, 2024.

[9] Yuqian Wu, Yuhong Peng, Jiapeng Yu, and Raymond Lee. Mas4poi: a multi-agents collaboration system for next poi recommendation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 356–367. Springer, 2025.

[10] Dongsheng Wang, Shen Gao, Chengrui Huang, Yuxi Huang, Ruixiang Feng, and Shuo Shang. Tool4poi: A tool-augmented LLM framework for next poi recommendation, 2025.

[11] Xinye Wanyan, Danula Hettiachchi, Chenglong Ma, Ziqi Xu, and Jeffrey Chan. Temporal-aware user behaviour simulation with large language models for recommender systems. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, CIKM '25, page 5335–5339. ACM, November 2025.

[12] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. In *European Conference on Information Retrieval*, pages 364–381. Springer, 2024.

[13] Chen Cheng, Haiqin Yang, Michael R Lyu, and Irwin King. Where you

like to go next: Successive point-of-interest recommendation. In *IJCAI*, volume 13, pages 2605–2611, 2013.

[14] Jing He, Xin Li, Lejian Liao, Dandan Song, and William Cheung. Inferring a personalized next point-of-interest recommendation model with latent behavior patterns. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

[15] Dejiang Kong and Fei Wu. Hst-lstm: A hierarchical spatial-temporal long-short term memory network for location prediction. In *Ijcai*, volume 18, pages 2341–2347, 2018.

[16] Ke Sun, Tieyun Qian, Tong Chen, Yile Liang, Quoc Viet Hung Nguyen, and Hongzhi Yin. Where to go next: Modeling long-and short-term user preferences for point-of-interest recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 214–221, 2020.

[17] Pengpeng Zhao, Anjing Luo, Yanchi Liu, Jiajie Xu, Zhixu Li, Fuzhen Zhuang, Victor S Sheng, and Xiaofang Zhou. Where to go next: A spatio-temporal gated network for next poi recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 34(5):2512–2524, 2020.

[18] Jie Feng, Yong Li, Chao Zhang, Funing Sun, Fanchao Meng, Ang Guo, and Depeng Jin. Deepmove: Predicting human mobility with attentional recurrent networks. In *Proceedings of the 2018 world wide web conference*, pages 1459–1468, 2018.

[19] Yuxia Wu, Ke Li, Guoshuai Zhao, and Xueming Qian. Personalized long-and short-term preference learning for next poi recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 34(4):1944–1957, 2020.

[20] Yingtao Luo, Qiang Liu, and Zhaocheng Liu. Stan: Spatio-temporal attention network for next location recommendation. In *Proceedings of the web conference 2021*, pages 2177–2185, 2021.

[21] Alif Al Hasan and Md. Musfique Anwar. Seaget: Seasonal and active hours guided graph enhanced transformer for the next poi recommendation. *Array*, 26:100385, 2025.

[22] Chao Song, Zheng Ren, and Li Lu. Integrating personalized spatio-temporal clustering for next poi recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(12):12550–12558, Apr. 2025.

[23] Zhaobo Wang, Yanmin Zhu, Haobing Liu, and Chunyang Wang. Learning graph-based disentangled representations for next poi recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pages 1154–1163, 2022.

[24] Song Yang, Jiamou Liu, and Kaiqi Zhao. Getnext: Trajectory flow map enhanced transformer for next poi recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on research and development in information retrieval*, pages 1144–1153, 2022.

[25] Xiaodong Yan, Tengwei Song, Yifeng Jiao, Jianshan He, Jiaotuan Wang, Ruopeng Li, and Wei Chu. Spatio-temporal hypergraph learning for next poi recommendation. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*, pages 403–412, 2023.

[26] Xunan Dong, Jun Zeng, Lin Zhong, Haoran Tang, Junhao Wen, and Min Gao. Fedhgs: A federated point-of-interest recommendation method based on heterogeneous graph semantics. *IEEE Transactions on Services Computing*, 2025.

[27] Wei Zhou, Cheng Fu, Chunyan Sang, Min Gao, and Junhao Wen. Next poi recommendation based on graph convolutional networks and multiple context-awareness. *IEEE Transactions on Services Computing*, 18(1):302–313, 2025.

[28] Hongli Zhou, Zhihao Jia, Haiyang Zhu, and Zhizheng Zhang. Cllp: Contrastive learning framework based on latent preferences for next poi recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1473–1482, 2024.

[29] Jiajie Chen, Yu Sang, Peng-Fei Zhang, Jiaan Wang, Jianfeng Qu, and Zhixu Li. Enhancing long-and short-term representations for next poi recommendations via frequency and hierarchical contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 11472–11480, 2025.

[30] Lin Chen, Peipei Wang, Xiaohui Han, and Lijuan Xu. Multi-relational variational contrastive learning for next poi recommendation. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[31] Yifan Wang and Qi Jiang. Enhancing next poi recommendation via multi-graph modeling and multi-granularity contrastive learning. *Journal of King Saud University Computer and Information Sciences*, 37(10):343, 2025.

[32] Qing Zhang, Xiao Huang, Qi Zhang, Wei Zhou, and Junhao Wen. Disentangling long-and short-term preferences based on latent commonalities enhancement for next point-of-interest recommendation. *Expert Systems with Applications*, page 127850, 2025.

[33] Shijie Liu, Ruixin Ding, Weihai Lu, Jun Wang, Mo Yu, Xiaoming Shi, and Wei Zhang. Coherency improved explainable recommendation via large language model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12201–12209, 2025.

[34] Jesse Harte, Wouter Zorgdrager, Panos Louridas, Asterios Katsifodimos, Dietmar Jannach, and Marios Fragkoulis. Leveraging large language models for sequential recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 1096–1102, 2023.

[35] Peiyu Hu, Wayne Lu, and Jia Wang. From ids to semantics: A generative framework for cross-domain recommendation with adaptive semantic tokenization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 14874–14882, 2026.

[36] Weihai Lu and Li Yin. Dmmd4sr: Diffusion model-based multi-level multimodal denoising for sequential recommendation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 6363–6372, 2025.

[37] Xiaoxi Cui, Weihai Lu, Yu Tong, Yiheng Li, and Zhejun Zhao. Multi-modal multi-behavior sequential recommendation with conditional diffusion-based feature denoising. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1593–1602, 2025.

[38] Xiaoxi Cui, Weihai Lu, Yu Tong, Yiheng Li, and Zhejun Zhao. Diffusion-based multi-modal synergy interest network for click-through rate prediction. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 581–591, 2025.

[39] Zhuoyang Liu and Weihai Lu. Mdn: Modality decomposition network for multimodal recommendation. In *Proceedings of the 2025 International Conference on Multimedia Retrieval*, pages 871–879, 2025.

[40] Minghao Mo, Weihai Lu, Qixiao Xie, Zikai Xiao, Xiang Lv, Hong Yang, and Yanchun Zhang. One multimodal plugin enhancing all: Clip-based pre-training framework enhancing multimodal item representations in recommendation systems. *Neurocomputing*, 637:130059, 2025.

[41] Chenhui Li and Weihai Lu. Decoding the market's pulse: Context-enriched agentic retrieval augmented generation for predicting post-earnings price shocks. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3055–3073, 2026.

[42] Penglong Zhai, Jie Li, Fanyi Di, Yue Liu, Yifang Yuan, Jie Huang, Peng Wu, Sicong Wang, Mingyang Yin, Tingting Hu, Yao Xu, and Xin Li. Cognitive-aligned spatio-temporal large language models for next point-of-interest prediction, 2025.

[43] Dongsheng Wang, Yuxi Huang, Shen Gao, Yifan Wang, Chengrui Huang, and Shuo Shang. Generative next poi recommendation with semantic id. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 2904–2914, 2025.

[44] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45(1):129–142, 2014.

[45] Eunjoon Cho, Seth A Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090, 2011.

[46] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 811–820, 2010.

[47] Shanshan Feng, Xutao Li, Yifeng Zeng, Gao Cong, Yeow Meng Chee, and Quan Yuan. Personalized ranking metric embedding for next new poi recommendation. In *IJCAI*, volume 15, pages 2069–2075, 2015.

[48] Joeun Kim, Youngjin Seo, Yeonsoo Kim, Junhyeok Kang, Jeeho Shin, Patara Trirat, and Jae-Gil Lee. Large language models are zero-shot point-of-interest recommenders. *Data Mining and Knowledge Discovery*, 39(6):80, 2025.