Accepted manuscript of an article published in *Knowledge-Based Systems*, vol. 354, 117234, 2026.
Published version: https://doi.org/10.1016/j.knosys.2026.117234.
© 2026. This manuscript version is made available under the CC BY-NC-ND 4.0 license, https://creativecommons.org/licenses/by-nc-nd/4.0/.

RALLM-STAR: Spatial-Temporal Aligned Retrieval-Augmented LLM for Zero-shot Next POI Recommendation

Kunrong Li[a], Kwan Hui Lim[b,*]

[a]*Information Systems Technology and Design, Singapore University of Technology and Design, 487372, Singapore*
[b]*School of Business, Singapore University of Social Sciences, 599494, Singapore*

ARTICLE INFO

Keywords:
Large Language Models
Next POI Recommendation
Retrieval Augmented Generation
Gaussian Mixture Modeling

ABSTRACT

Next Point-of-Interest (POI) recommendation aims to predict a user's future destination by uncovering latent mobility patterns from historical trajectories. While traditional deep learning models suffer from high training costs and poor generalization, Large Language Models (LLMs) offer a promising zero-shot alternative, yet often struggle with hallucinations and a lack of spatial grounding. To bridge this gap, we propose RALLM-STAR, a retrieval-augmented LLM framework designed to inject rigorous spatial-temporal context into the reasoning process. First, we introduce the Spatial-Temporal Alignment Retriever (STAR), which captures the duality of user intent by modeling both short-term interests and long-term habits, integrated with a Gaussian Mixture Model to ensure distributional spatial consistency. Second, to enforce fine-grained sequential realism, we employ a Geographical Distance Reranker (GDR) that prioritizes candidates exhibiting high path-level alignment with the recent movements of the user. Finally, we incorporate an Agentic LLM Rectifier (ALR), a self-reflective module that iteratively verifies and corrects generation errors to ensure robust outputs. Extensive experiments on three real-world Foursquare datasets, plus large-scale evaluation on Gowalla, demonstrate that RALLM-STAR achieves substantial accuracy gains, outperforming existing zero-shot LLM-based approaches and remaining competitive with recent supervised graph/hypergraph recommenders without additional training. The code is released at https://github.com/LKRcrocodile/RALLM-STAR.

1. Introduction

Next Point-of-Interest (POI) recommendation remains a pivotal yet challenging task in spatial-temporal data mining. The primary objective is to predict the next likely destination of a user using historical trajectories that encode rich, latent patterns of check-in behavior and personal preferences [45, 21]. Accurate next-POI prediction not only enhances user experiences in Location-Based Social Networks (LBSNs), but also offers broader social benefits, ranging from optimized traffic management and urban planning to public health monitoring and disaster response. Consequently, the task has garnered extensive research attention, fostering diverse models aimed at deciphering the complexities of human mobility.

Traditionally, deep learning approaches have dominated this field. Recurrent Neural Networks (RNNs), such as LSTMs and GRUs, have been widely adopted to model sequential dependencies in check-in data [41, 38, 47, 11, 20]. Concurrently, graph-based models have been utilized to capture global transition patterns and mitigate data sparsity by exploiting the structural topology of mobility networks [36, 43, 34]. More recently, Transformer architectures have gained prominence, treating check-in sequences akin to natural language to capture long-range contextual dependencies [31, 42, 7, 23, 3, 18]. However, despite their efficacy, these fully supervised paradigms typically require massive labeled datasets and extensive training. Furthermore, they often lack zero-shot generalization capabilities, struggling to adapt to new users or regions without retraining. This limitation has catalyzed interest in alternative paradigms that can predict mobility patterns in a training-free, zero-shot manner.

The emergence of Large Language Models (LLMs) offers a promising avenue to address these limitations. Recent studies have explored the integration of LLMs in POI recommendations to take advantage of their vast semantic knowledge base and reasoning capabilities. For example, frameworks such as LLMMove [5] and LLM-Mob [35] utilize carefully engineered prompts to transform raw check-in histories into narrative descriptions, allowing LLM to infer mobility intent. Others, such as LLMZS [1], rely on in-context learning by providing examples of inputs and expected outputs. Nevertheless, existing zero-shot LLM approaches face significant hurdles. A key challenge is the *context gap*: simply prompting an LLM with a user's raw history is often suboptimal. Users' intentions oscillate dynamically between ingrained long-term habits and transient short-term goals, while standard prompting fails to distinguish these nuances. Moreover, LLMs are highly sensitive to the quality and ordering of retrieved context [28, 12], and without rigorous spatial grounding, they are prone to hallucinating plausible but geographically invalid recommendations.

Beyond direct prompting, recent studies have explored retrieval augmentation for recommendation,

*Corresponding author

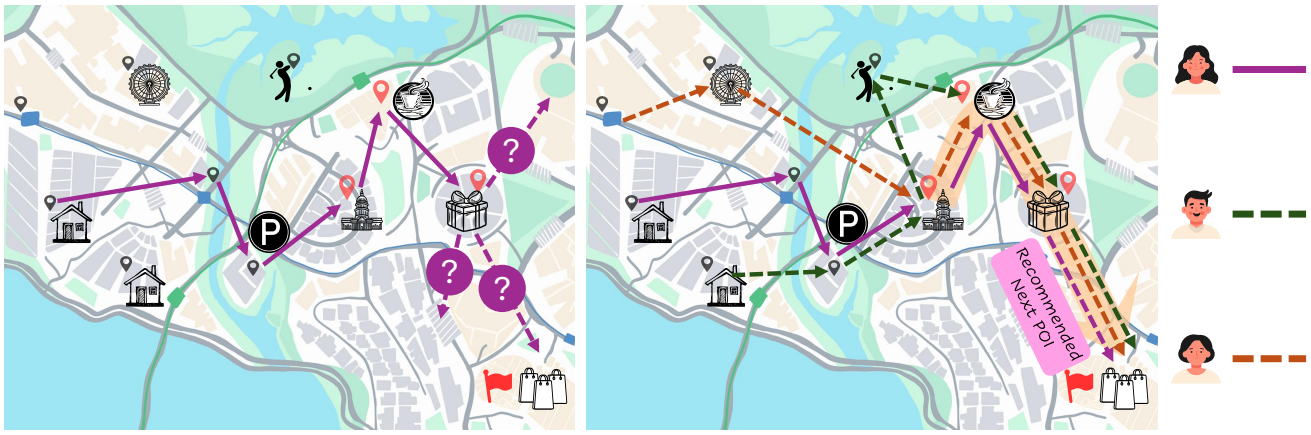


Figure 1: Illustration of the RALLM-STAR framework. Traditional methods struggle with sparse information (purple). Our approach retrieves geographically and semantically aligned trajectories (green and orange) using probabilistic modeling, and refines them to provide the LLM with a supportive, grounded context for predicting the next POI.

trajectory retrieval for mobility prediction, and iterative self-refinement for improving LLM outputs. Recommendation-oriented RAG methods enrich LLMs with external item, collaborative, or knowledge-based evidence [33, 25], while trajectory-based methods retrieve informative historical mobility sequences to support next-POI prediction [19, 24]. Meanwhile, self-refinement methods improve initial LLM generations through iterative feedback and revision [26, 29]. These advances motivate an important question for zero-shot next-POI recommendation: how to align retrieved historical evidence with the target user's mobility behavior at multiple spatial-temporal resolutions and organize such evidence effectively for LLM inference.

Motivated by this challenge, we propose an enhanced **R**etrieval **A**ugmented **L**arge **L**anguage **M**odel framework for next POI recommendation with **S**patial-**T**emporal **A**ligned **R**etrieval, termed **RALLM-STAR**. RALLM-STAR adopts a multi-stage RAG pipeline that progressively aligns historical mobility evidence with the target user's behavioral intent and movement patterns before LLM prediction. First, to bridge the context gap by providing the LLM with high-quality, grounded evidence, we propose the **Spatial-Temporal Alignment Retriever (STAR)**. The fundamental objective of STAR is to query a database of historical mobility sequences to identify and retrieve trajectories that exhibit similar semantic and kinematic patterns to the target user. To achieve this with high precision, STAR goes beyond simple similarity matching by employing a dual-stream architecture that explicitly disentangles short-term interests from long-term habits. Furthermore, STAR incorporates Gaussian Mixture Models (GMM) to ground the retrieval process in a probabilistic spatial reality, ensuring that retrieved contexts are both semantically relevant and distributionally plausible. Second, to refine the retrieved context for the LLM, we introduce the **Geographical Distance Reranker (GDR)**. While STAR ensures alignment at a distributional level, GDR focuses on the fine-grained sequential order of movement. By utilizing Decaying Weighted Dynamic Time Warping (DWDTW), GDR prioritizes support trajectories that mirror the user's recent path, incorporating a recency bias to align with the LLM's sensitivity to prompt ordering. Finally, acknowledging that LLMs can still produce format errors or logical inconsistencies in a single generation pass, we incorporate an **Agentic LLM Rectifier (ALR)**. Inspired by agentic workflows [30], ALR acts as a self-reflective quality assurance layer, which iteratively verifies and corrects the output to ensure robustness and compliance with task constraints. Together, these components form a mobility-specific multi-level alignment pipeline, progressing from short- and long-term intent alignment and spatial-distribution alignment to path-level evidence prioritization and structured output verification.

We extensively validate RALLM-STAR on three real-world Foursquare datasets and a large-scale Gowalla benchmark. Our results demonstrate that by combining probabilistic retrieval, sequential spatial reranking, and agentic rectification, our framework surpasses state-of-the-art zero-shot LLM baselines and competes closely with recent graph/hypergraph recommenders without any per-dataset training, and generalizes robustly across six LLM backbones.

2. Related Work

2.1. Next POI Recommendation

The next POI recommendation predicts the most likely POI a user will select next, based on their past check-ins, the density of surrounding POIs, and patterns in user behavior [45]. Classic neural network-based models marked a significant advancement in leveraging deep learning for recommendations. Differ from traditional latent factor methods such as LDA-based and matrix factorization-based models, they

model interactions between users, locations, and contextual information more comprehensively. The prominent examples include recurrent structures designed for sequence modeling [41, 38, 47, 11, 20] and variations of transformer-based models [42, 7, 10, 8, 23, 44, 3]. For sequential-based methods, such as ALSTM [11], LSTM networks are enhanced with an attention mechanism that selectively focuses on relevant historical check-ins, improving prediction accuracy by capturing spatial proximity and temporal dynamics. Similarly, MCRNN [20] integrates spatial-activity topics into a multi-task learning framework, combined with its novel Context Aware Recurrent Unit to dynamically fuse sequential dependencies and temporal regularity, effectively handling sparsity and enhancing user activity-location prediction. Transformer-based models show better performance in capturing complex contextual and sequential dependencies. For example, CTLE [23] employs a bidirectional Transformer encoder to dynamically generate context-specific embeddings for multi-functional locations, integrating temporal information and masked pre-training objectives to boost prediction accuracy. CFPRec [44] further models past, current, and inferred multi-step future preferences via a Transformer layer, leveraging the periodicity of user check-ins for holistic preference learning. Concurrently, graph and hypergraph-based models have become increasingly prominent in next-POI recommendation, with methods such as DCHL [15], CAMSHG [48], and MSAHG [22] achieving strong results by modeling multi-scale POI transition topologies and cross-user collaborative signals. However, these methods often overfit to limited relationships in training data and lack semantic understanding of the trajectories and generalization ability. Meanwhile, they tend to require substantial computational resources and time for training. Therefore, there is a critical need for utilizing pre-trained models for POI recommendation in a zero-shot paradigm.

2.2. LLM for Sequential Recommendation

LLMs have demonstrated remarkable capabilities across a variety of domains due to their strong language understanding and reasoning abilities [35, 1, 5, 9, 19, 17]. Recently, there has been growing interest in leveraging LLMs for sequential recommendation tasks. A number of methods have been proposed that adapt or fine-tune LLMs specifically for recommending sequences of items or POIs to users. Some approaches require supervised training on recommendation data, such as LLMSeqPrompt [9] and LLMTRSR [46], but these methods often demand even larger computational costs while achieving comparable performance with specialized recommendation models, especially when only limited data is available.

In contrast, zero-shot paradigms that harness the inherent knowledge and in-context learning of LLMs have achieved promising results. Methods such as LLM-Mob [35] and LLMMove [5] directly prompt pretrained language models for recommendation without further training or extensive labeled data, while LLMZS [1] leverages structured in-context examples for zero-shot next-location prediction. These methods demonstrate the potential of LLMs for flexible and training-free sequential recommendation. However, their effectiveness still depends heavily on the quality of the contextual information supplied to the LLM, motivating recent research on retrieval-augmented recommendation and trajectory-based context construction.

2.3. Retrieval-Augmented and Self-Refinement Methods

Retrieval-Augmented Generation (RAG) has emerged as an effective paradigm for grounding LLM outputs with external evidence. Generic RAG frameworks have progressively improved retrieval and generation through structured knowledge retrieval, speculative generation, and relevance correction, such as GraphRAG [4], Speculative RAG [37], and Corrective RAG [39]. More recently, RAG has been adapted to recommendation tasks. K-RagRec [33] retrieves structural information from external knowledge graphs to augment recommendation, while RALLRec+ [25] learns joint item representations from textual and collaborative semantics and further reranks retrieved items according to dynamic user preferences. These methods demonstrate the effectiveness of external evidence for recommendation. RALLM-STAR specifically focuses on mobility evidence by retrieving complete historical trajectories as behavioral exemplars and evaluating their relevance through mobility-specific temporal and geographical alignment.

Trajectory retrieval provides another closely related direction for next-POI recommendation. Li et al. [19] encode current and historical trajectories with an LLM encoder and select historical trajectories according to key-query cosine similarity, allowing trajectories from similar users to augment the current prediction. STrajRAG [24] further introduces a supervised auxiliary trajectory-selection task that incorporates heterogeneous similarities such as user and temporal information to retrieve trajectories likely to contain informative next-POI evidence. Compared with these trajectory retrieval strategies, RALLM-STAR performs multi-resolution alignment across temporal intent, spatial distribution, and recent movement paths, and uses the resulting ranking to organize the evidence presented to the LLM.

LLM self-refinement methods aim to improve an initial generation through feedback and subsequent revision. For example, Self-Refine [26] iteratively uses an LLM to provide feedback on and refine its own output, while Reflexion [29] enables language agents to exploit verbal feedback from previous attempts. In

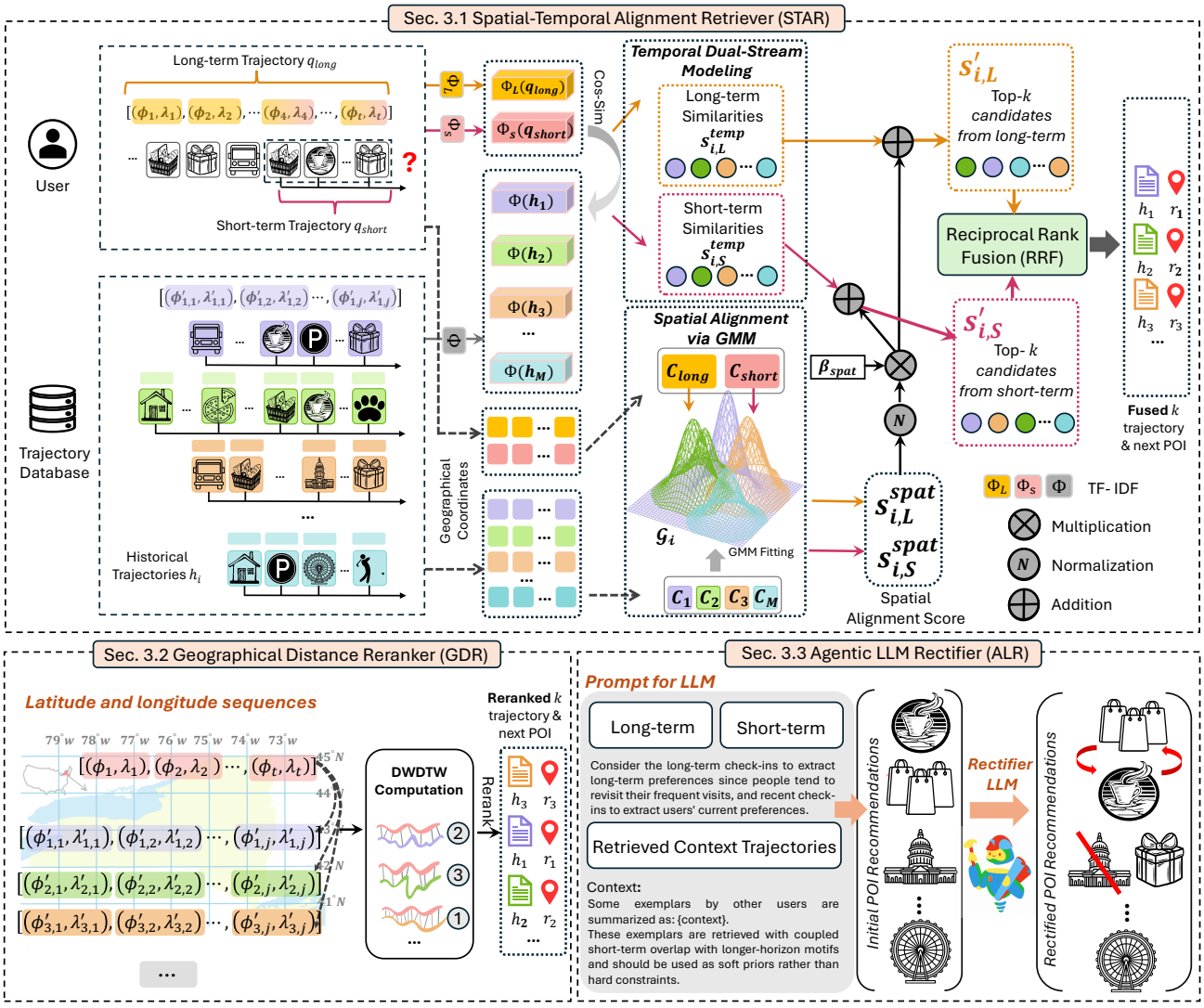


Figure 2: Illustration of the proposed RALLM-STAR framework for next POI recommendation.

RALLM-STAR, self-refinement is adapted as a task-specific verification mechanism for structured next-POI recommendation, where ALR verifies recommendation constraints and corrects the output using the spatial-temporally grounded evidence produced by STAR and GDR.

3. Methodology

3.1. Spatial-Temporal Alignment Retriever (STAR)

In personalized POI recommendation, user trajectories encode latent behavioral intents that manifest themselves as correlated spatial and temporal patterns. The fundamental premise is that *trajectories exhibiting similar semantic and spatial-temporal characteristics imply similar future intentions*. For instance, a user whose sequence includes *(bookstore → café)* likely expresses a "leisurely afternoon" intent. Retrieving analogous trajectories from other users thus provides powerful, contextually-grounded evidence for predicting their next destination. To this end, we propose to formalize this through a retrieval framework named Spatial-Temporal Alignment Retriever (STAR) with probabilistic modeling. It sources informative contexts from a large database of historical trajectories, $\mathcal{D} = \{(h_i, r_i)\}_{i=1}^{M}$, where $h_i = [p_{i,1}, \ldots, p_{i,T_i}]$ is a historical trajectory of visited POIs, and r_i is the corresponding ground-truth next POI. Given a query trajectory $q = [p_{q,1}, \ldots, p_{q,T_q}]$, the objective is to retrieve a set of historical trajectories $\mathcal{R}(q) \subset \mathcal{D}$ that maximize the posterior relevance probability:

$$\mathcal{R}(q) = \arg\max_{\mathcal{S} \subseteq \mathcal{D}} P(\mathcal{S} \mid q), \tag{1}$$

where $P(\mathcal{S} \mid q)$ reflects both semantic and spatial-temporal congruence. We encode both the query trajectory q and all database entries h_i into a common high-dimensional vector space using a TF-IDF encoder, denoted as $\Phi(\cdot)$. Specifically, each POI identifier p acts as a term within the global vocabulary $\mathcal{V}$ which contains

all POIs in the training trajectories. Formally, for each trajectory h_i, we have

$$\Phi(h_i) = \big[\mathrm{TF}(p, h_i) \cdot \log \frac{M}{1 + |\{h \in \mathcal{D} : p \in h\}|}\big]_{p \in \mathcal{V}}, \quad (2)$$

so that the model could simultaneously measure each POI's prevalence in the trajectory meanwhile emphasizes distinctive mobility patterns rather than universally popular POIs. Afterwards, the semantic similarity between the query embedding $\Phi(q)$ and the database embedding $\Phi(h_i)$ can be calculated as the cosine similarity:

$$s_i = \cos(\Phi(q), \Phi(h_i)) = \frac{\Phi(q)^\top \Phi(h_i)}{\|\Phi(q)\| \, \|\Phi(h_i)\|}. \quad (3)$$

Temporal Dual-Stream Modeling. Furthermore, to create a more nuanced retrieval set that better reflects user mobility patterns, we consider that a user's intent is not static but evolves over time. We enhance this semantic foundation by explicitly modeling the distinction between a user's transient, short-term interests and their ingrained, long-term habits. To address this, we propose a multi-faceted retrieval architecture that operates on dual temporal streams and combines selections with a spatial-probabilistic re-ranking mechanism. Instead of directly applying the database encoder to embed a fixed query, we propose to decompose our test user history into temporal segments. Specifically, for a query q, we distinguish its recent short-term segment q_{short} from its long-term history q_{long}. We then introduce two individual encoders Φ_S and Φ_L which are constructed using TF–IDF exclusively from this trajectory. This operation enables a personalized representation for each test user. Then, we compute two sets of embeddings for the two queries: a short-term query $\Phi_S(q_{short})$, and a long-term query $\Phi_L(q_{long})$, which enables the calculation of two parallel similarity scores that represent the temporal correlation between the query and candidate h_i:

$$s_{i,S}^{\mathrm{temp}} = \frac{\Phi_S(q_{short}) \cdot \Phi(h_i)}{\|\Phi_S(q_{short})\| \|\Phi(h_i)\|} \ \text{(Short-Term Temporal)} \quad (4)$$

$$s_{i,L}^{\mathrm{temp}} = \frac{\Phi_L(q_{long}) \cdot \Phi(h_i)}{\|\Phi_L(q_{long})\| \|\Phi(h_i)\|} \ \text{(Long-Term Temporal)} \quad (5)$$

Here, s^{temp} captures the overlap of activity patterns across different time horizons, serving as the temporal component.

Spatial Alignment via Gaussian Mixture Modeling. While TF-IDF captures *what* POIs are visited, it ignores *where* they are located. To incorporate geographical realism, we model the spatial distribution of each candidate trajectory independently. Specifically, for trajectory h_i, let the set of geographical coordinates be

$$C_i = \{\mathbf{c}_{i,t} = \mathrm{geo}(p_{i,t}) \in \mathbb{R}^2 \mid p_{i,t} \in h_i\}. \quad (6)$$

where $\mathrm{geo}(\cdot)$ is a deterministic mapping that converts each POI identifier $p_{i,t}$ into its precise 2D corresponding geographical coordinates (longitude and latitude). To capture the underlying spatial patterns including clustering, directional bias, and regional spread, we propose to fit a GMM $\mathcal{G}_i$ to C_i with G components, which is parameterized by

$$\mathcal{G}_i(C_i) = \sum_{g=1}^{G} \pi_{i,g} \, \mathcal{N}(\mathbf{c}; \boldsymbol{\mu}_{i,g}, \boldsymbol{\Sigma}_{i,g}), \quad (7)$$

where $\pi_{i,g}$ are the mixture weights satisfying $\sum_g \pi_{i,g} = 1$, $\boldsymbol{\mu}_{i,g}$ are component means, and $\boldsymbol{\Sigma}_{i,g}$ are covariance matrices capturing spatial spread. Taking the query's short-term coordinates C_{short} as an example, we define the spatial alignment score by calculating how well the query's locations fit into the candidate's spatial distribution:

$$\mathcal{L}_i = \prod_{\mathbf{c} \in C_{short}} P(\mathbf{c} \mid \mathcal{G}_i) = \prod_{\mathbf{c} \in C_{short}} \sum_{g=1}^{G} \pi_{i,g} \, \mathcal{N}(\mathbf{c}; \boldsymbol{\mu}_{i,g}, \boldsymbol{\Sigma}_{i,g}), \quad (8)$$

and take log-average as a normalized spatial alignment score to mitigate numerical underflow and unstable gradients:

$$s_{i,S}^{\mathrm{spat}} = \frac{1}{|C_{short}|} \log \mathcal{L}_i = \frac{1}{|C_{short}|} \sum_{\mathbf{c} \in C_{short}} \log P(\mathbf{c} \mid \mathcal{G}_i). \quad (9)$$

This score s^{spat} measures the distributional spatial consistency, serving as the spatial component of STAR. It can be interpreted as estimating the expected log-density $\mathbb{E}_{\mathbf{c} \sim C_{short}}[\log \mathcal{G}_i(\mathbf{c})]$, which formally measures how probable the query's spatial behavior is under the generative distribution of h_i. After min-max normalization of the spatial scores to $\hat{s}^{\mathrm{spat}} \in [0, 1]$, we refine the temporal similarities to obtain the final STAR ranking scores:

$$s'_{i,S} = s_{i,S}^{\mathrm{temp}} + \beta_{\mathrm{spat}} \, \hat{s}_{i,S}^{\mathrm{spat}}, \quad (10)$$

$$s'_{i,L} = s_{i,L}^{\mathrm{temp}} + \beta_{\mathrm{spat}} \, \hat{s}_{i,L}^{\mathrm{spat}}, \quad (11)$$

where β_{spat} is a balancing coefficient controlling the influence of the GMM-based spatial alignment.

Finally, the two ranked lists produced by the short-term and long-term streams are merged into a single, definitive ranking. We employ Reciprocal Rank Fusion (RRF) [2] to synthesize the results, giving preference to trajectories that rank highly in either or both streams. Let $\mathcal{I}_S$ and $\mathcal{I}_L$ be the sets of indices for the top-k candidates from the short-term and long-term

lists, respectively. The final composite score for each trajectory h_i is calculated as:

$$\text{Score}(i) = \mathbb{I}[i \in \mathcal{I}_S] \cdot \frac{\beta_{temp}}{\text{rank}(s'_{i,S}) + \delta} + \mathbb{I}[i \in \mathcal{I}_L] \cdot \frac{1 - \beta_{temp}}{\text{rank}(s'_{i,L}) + \delta} \tag{12}$$

where $\text{rank}(s'_{i,S/L})$ denotes the ranking position of trajectory i sorted by the corresponding score $s'_{i,S/L}$ in descending order. Here, $\mathbb{I}[\cdot]$ is the indicator function, δ is a smoothing constant (e.g., 1), and $\beta_{temp} \in [0, 1]$ is a hyperparameter balancing the two temporal streams. The final top-$\hat{k}$ ($\hat{k} < k$) historical trajectories according to this final score are selected to construct a context that is semantically relevant, temporally balanced, and geographically plausible.

Bayesian Interpretation. From a probabilistic standpoint, STAR approximates the posterior relevance

$$P(h_i \mid q) \propto \underbrace{P(q \mid h_i)_{\text{temporal}}}_{\text{TF-IDF Similarity}} \cdot \underbrace{P(q \mid \mathcal{G}_i)_{\text{spatial}}}_{\text{GMM Likelihood}}. \tag{13}$$

where the first term is implicitly modeled by the cosine similarity in Eq. (3), and the second term by the GMM likelihood in Eq. (9).

3.2. Geographical Distance Reranker (GDR)

After STAR identifies candidates that are semantically relevant and ensures alignment at a distributional spatial level, we obtain a set of candidates whose overall trajectory is consistent with the user's recent spatial context. However, while STAR effectively aligns candidates based on temporal patterns (s^{temp}) and distributional spatial regions (s^{spat}), it does not explicitly enforce the fine-grained **sequential order** of the path. A user's immediate intent should be analyzed via both the region they inhabit as well as the path they are currently following. To address this, we introduce the Geographical Distance Reranker (GDR). GDR is designed to complement STAR by re-ranking its candidates based on the detailed POI correspondence to the query. This re-ranking is crucial, as LLMs can be sensitive to the ordering and quality [28, 12] of support examples within the prompt. By prioritizing trajectories that best match the query user's recent movement path and order, GDR ensures the LLM receives the most pertinent contextual examples.

Specifically, let the recent query trajectory with length T_q be denoted as $q = [p_{q,1}, \ldots, p_{q,T_q}]$. Each POI $p_{q,t}$ is mapped to its associated geographical coordinate $\mathbf{c}_{q,t} \in \mathbb{R}^2$. To be precise, we denote the coordinate vector as $\mathbf{c}_{q,t} = (\phi_{q,t}, \lambda_{q,t})$, where ϕ and λ represent latitude and longitude, respectively. This forms the query coordinate sequence $C_q = [\mathbf{c}_{q,1}, \ldots, \mathbf{c}_{q,T_q}]$. Similarly, for a certain candidate support trajectory h_i retrieved by STAR with length T_i, we extract its coordinate sequence $C_i = [\mathbf{c}_{i,1}, \ldots, \mathbf{c}_{i,T_i}]$, where each candidate point is defined as $\mathbf{c}_{i,j} = (\phi'_{i,j}, \lambda'_{i,j})$ for $j \in [1, T_i]$.

To quantify the path-level spatial alignment between the user's current movement and each candidate, we propose a Decaying Weighted Dynamic Time Warping (DWDTW) distance based on real-world geospatial separation. It is notable that user trajectories can vary in length and are often temporally misaligned due to differences in their pace or the number of locations visited. Therefore, traditional point-to-point metrics such as Euclidean distance would not adequately handle these temporal shifts or differences in sequence length. Dynamic Time Warping (DTW) [27, 13] addresses this by finding an optimal alignment between the two sequences and minimizes the cumulative spatial cost over all pairs of matched points.

To emphasize the importance of recent visited locations as they are more predictive of immediate intent, we implement a recency bias with exponentially decaying weights, $\omega_t = \rho^{T_q - t}$ for $\rho \in (0, 1)$, where t is the time step index in the query. An alignment path is a sequence of index pairs (t, j) that matches points from q to h_i monotonically. The DWDTW alignment cost is computed as:

$$\text{DWDTW}(q, h_i) = \min \sum_{t=1}^{T_q} \sum_{j=1}^{T_i} \omega_t \cdot d(\mathbf{c}_{q,t}, \mathbf{c}_{i,j}), \tag{14}$$

where $d(\mathbf{c}_{q,t}, \mathbf{c}_{i,j})$ is the Haversine distance calculated using the latitudes and longitudes (ϕ, λ) of the matched points. Lower DWDTW values indicate stronger path-level alignment, particularly at the most recent steps. With DWDTW, even if users may take slightly different paths to similar destinations, we could still identify the most supportive trajectories. We compute the DWDTW value for all retrieved candidates h_i and rerank them in ascending order accordingly. Then, they are placed into the prompt as contextual support for the LLM. Lower DWDTW values indicate stronger path-level alignment, particularly at the most recent steps. With DWDTW, even if users may take slightly different paths to similar destinations, we could still identify the most supportive trajectories.

We compute the DWDTW value for all retrieved candidate coordinate sequences C_i and rerank them in ascending order accordingly. Then, they are placed into the prompt as contextual support for the LLM. Furthermore, we enable the LLM to be explicitly instructed to use its own judgment on whether to utilize this context information as examplars, thereby avoiding over-reliance on retrieved information:

Context Information for Prompting the LLM after STAR and GDR

Context: Some exemplars by other users are summarized as: `retieved context`.
These exemplars are retrieved with coupled short-term overlap with longer-horizon motifs and should be used as soft priors rather than hard constraints.

To this end, our GDR further enhances STAR that injects comprehensive spatial awareness into the prompt construction process. STAR first acts as an efficient retriever, identifying candidates from the correct spatial regions (distributional alignment), while GDR then acts as a precise re-ranker, promoting candidates with the most similar movement paths (sequential alignment). This ensures that the most geographically and sequentially plausible examples are presented most prominently, guiding the LLM to generate POI recommendations that are more likely to follow realistic movement patterns. Importantly, GDR is not merely an external pre-filter. It determines the order of the retrieved exemplars that are actually exposed to the LLM. Since LLMs are known to be sensitive to the quality and ordering of prompt evidence [28, 12], this reranking step directly affects how limited prompt capacity is allocated. We validate this connection empirically through a prompt-order sensitivity test in Section 4.

3.3. Agentic LLM Rectifier (ALR)

To further enhance reliability and compliance in our recommendation pipeline, we incorporate an Agentic LLM Rectifier (ALR) inspired by recent developments in agentic Retrieval-Augmented Generation (RAG) [30]. While the preceding components including probabilistic retrieval and spatial reranking could effectively retrieve and contextualize relevant user trajectory data, the inherent complexity of structured recommendation tasks often necessitates an additional verification and refinement layer to enforce critical constraints such as output formatting, recommendation uniqueness, result completeness, and alignment with inferred user intent. This observation is grounded in empirical findings that single-pass LLM outputs frequently exhibit inconsistencies or subtle deviations from task specifications, particularly in constrained settings where multiple interdependent requirements must be simultaneously satisfied [32, 6]. By adopting an agentic approach, the model is prompted to assume an active and reflective role in self-assessment and iterative revision, thereby mirroring human-in-the-loop postprocessing [14] and quality assurance workflows in a fully automated manner. This process leverages the inherent reasoning and meta-cognitive capabilities of LLMs for multi-turn correction, enabling the system to detect inconsistencies, validate constraint satisfaction, and refine outputs iteratively, resulting in substantially more robust and reliable system behavior. Furthermore, the agentic paradigm introduces interpretability benefits, as each correction step provides explicit rationale for output modifications, facilitating downstream debugging and system auditing.

Specifically, the agentic rectification prompt is structured as follows:

Agentic LLM Rectifier Prompt

A user has completed the following task:
<Prompt of prior LLM>
<Response of prior LLM>
Your task:
Carefully review the answer above. Evaluate whether it satisfies all requirements. If it is correct, concise, and well-formatted, reproduce the answer. If you find any issues (formatting, duplicates, insufficient recommendations, irrelevant POI IDs, unsatisfactory reasoning, etc), revise the answer to fully meet the requirements. The result should be ONLY the final (possibly revised) with the SAME format as the given response, which is a JSON object containing your "recommendation" and "reason".

Therefore, the agentic rectifier LLM strategy could rigorously detects and corrects errors that may otherwise propagate to the final output, ensuring that the final response strictly adheres to all specified constraints and quality thresholds. This additional verification step is particularly critical in recommendation systems where constraint violations directly impact user experience and system credibility.

4. Experiments

4.1. Experimental Setup

Datasets. We conduct experiments on three Foursquare datasets [40] representing Singapore (SIN), New York City (NYC), and Phoenix (PHO). For preprocessing and ensuring data quality, we filter out POIs with fewer than 10 interactions. We also remove users that have less than 5 trajectories, and discard trajectories with no more than 3 POIs. The interaction records are chronologically sorted for each user, and 80% of the data is used for database construction, while the remaining is for testing following [44]. Each long-term sequence comprises all qualifying daily trajectories from the same user that fall within the seven days preceding the short-term sequence. Since our method is zero-shot, no validation set is required. To assess RALLM-STAR under a substantially larger user and POI space, we further conduct experiments on the large-scale Gowalla dataset. Statistical information of datasets are given in Table 1.

Table 1
Statistical information of the datasets.

	#User	#POI	#Check-in	#Density
PHO	2,946	7,247	47,980	0.22%
NYC	16,387	56,252	511,431	0.06%
SIN	8,648	33,712	355,337	0.12%
Gowalla	30,312	404,884	1,728,535	0.014%

Baselines. We compare our proposed RALLM-STAR with several representative training-based approaches and zero-shot LLM methods, which are introduced as follows. (1) ALSTM [11] leverages an attention mechanism to enable the spatial-temporal LSTM to assign different levels of importance to relevant historical check-ins. (2) MCRNN [20] utilizes a multi-task learning approach, modeling spatial-activity topics as latent factors to effectively capture both user activities and location preferences. (3) CTLE [23] is a bi-directional Transformer model that considers contextual neighbors within trajectories and incorporates temporal information to enhance location embeddings. (4) CFPRec [44] integrates attention mechanisms with LSTM to model both past and present user preferences, employing a two-layer attention structure to predict users' multi-step future interests. (5) LLMMob [35] encodes temporal information within human mobility sequences, using these as prompts for LLMs. (6) LLMMove [5] directs the LLM to balance long-term and recent preferences along with sequential transitions. (7) LLMZS [1] creates a structured prompt that incorporates data from users' past and contextual visits. (8) RALLM-POI [16] is the prior conference version that represents the initial attempt to apply the RAG-based technique to recommend the user's next visit intent. (9) DCHL [15] is a hypergraph-based next-POI model that captures dynamic collaborative signals via dual-channel hypergraph learning. (10) CAMSHG [48] employs a context-aware multi-scale hypergraph to model complex user-POI interactions. (11) MSAHG [22] introduces multi-scale adaptive hypergraph learning for next POI recommendation.

Evaluation Metrics. For evaluation, we employ two widely used metrics in next POI recommendation: Hit Ratio (HR) and Normalized Discounted Cumulative Gain (NDCG). HR@K measures the proportion of times the target POI appears among the top-K recommended POIs, thus evaluating whether positive items are present in the top recommendations. In contrast, NDCG@K takes into account the ranking positions of positive POIs within the top-K results, assigning higher rewards when positive items appear closer to the top. In this work, we report HR and NDCG at K = 5 and K = 10 for each method to ensure a fair comparison.

Implementation Details. We employ GPT-4o-mini as the underlying LLM for all zero-shot methods (LLMMob, LLMMove, LLMZS, RALLM-POI, and RALLM-STAR) to ensure fair comparison. All methods are configured with the identical API endpoint and model identifier; any performance difference is therefore attributable to methodology rather than the LLM backbone. The temperature and top-p of the LLM is set at 0.01 to reduce randomness in responses. To facilitate the generalization ability of our method, we applied fixed hyperparameters across all three datasets, and use ablation experiments for the choices. The λ is set to 0.8 and we retrieve the k=10 candidates for both long and short term queries, with $\hat{k}$=3 as final context. We empirically fit G=3 clusters for the GMM modeling in STAR.

4.2. Overall Performance

Table 2 presents the overall performance comparison on the PHO, NYC, and SIN datasets. We evaluate RALLM-STAR against seven representative training-based methods (ALSTM [11], MCRNN [20], CTLE [23], CFPRec [44], DCHL [15], CAMSHG [48], and MSAHG [22]), three state-of-the-art zero-shot LLM baselines (LLMMob [35], LLMMove [5], LLMZS [1]), and our prior conference version RALLM-POI [16]. Across all metrics and datasets, RALLM-STAR consistently achieves the best performance among zero-shot methods, demonstrating the effectiveness of integrating probabilistic retrieval, spatial reranking, and agentic self-rectification.

Comparison with training-based models. Traditional sequence modeling approaches (ALSTM, MCRNN) exhibit substantially lower performance across all metrics. These models rely on recurrent state transitions over individual user histories and are therefore unable to capture cross-user collaborative signals or global POI transition patterns, which is particularly problematic in sparse and geographically diverse environments. Transformer-based models (CTLE, CFPRec) provide noticeable improvements over recurrent architectures by modeling long-range temporal dependencies via self-attention, yet still treat each user's trajectory in isolation without explicitly encoding structural relationships across the full POI graph. Recent graph and hypergraph models (DCHL, CAMSHG, MSAHG) achieve substantially stronger results by explicitly modeling the structural topology of POI transition networks and propagating collaborative signals across users. On PHO, CAMSHG and MSAHG both reach HR@5 = 0.5789, a 3.7× improvement over ALSTM; on SIN, CAMSHG achieves HR@5 = 0.4462, which surpasses CFPRec by a large margin. Graph can aggregate evidence from all users who visited either POI, yielding robust transition estimates even when a specific POI pair rarely co-occurs in a single user's history. When compared to CFPRec, RALLM-STAR

Table 2
Performance comparison in HR@K and NDCG@K on three datasets. "H" stands for HR and "N" stands for NDCG. The methods are grouped into training-based and zero-shot methods. Within each paradigm, the best result is in **bold** and the second-best is in *italics*.

Data	Metric	*Training-based Methods*							*Zero-shot Methods*				
		ALSTM	MCRNN	CTLE	CFPRec	DCHL	CAMSHG	MSAHG	LLMMob	LLMMove	LLMZS	RALLM-POI	RALLM-STAR
PHO	H@5	0.1579	0.1905	0.2632	0.3421	*0.5526*	**0.5789**	**0.5789**	0.3158	0.3157	0.3421	*0.5263*	**0.5526**
	H@10	0.2377	0.2726	0.3605	0.4253	**0.6842**	0.6316	*0.6579*	0.5789	0.6051	0.5526	*0.6842*	**0.7105**
	N@5	0.1033	0.1264	0.1995	0.2432	**0.4052**	*0.4015*	0.3976	0.2355	0.2479	0.2520	*0.3505*	**0.3620**
	N@10	0.1385	0.1617	0.2068	0.2730	**0.4457**	0.4171	*0.4245*	0.3193	0.3434	0.3191	*0.3989*	**0.4121**
NYC	H@5	0.1667	0.1835	0.2421	0.2734	0.3990	*0.4007*	**0.4057**	0.2465	0.2470	0.2020	*0.2554*	**0.3529**
	H@10	0.2031	0.2397	0.3205	0.3306	**0.4975**	0.4725	*0.4942*	0.4071	0.4023	0.3723	*0.4190*	**0.4508**
	N@5	0.0912	0.1036	0.1513	0.1588	0.2861	*0.2984*	**0.3002**	0.1571	0.1565	0.1285	*0.1599*	**0.2479**
	N@10	0.1638	0.1870	0.1841	0.1834	0.3175	*0.3215*	**0.3286**	0.2107	*0.2132*	0.1838	*0.2132*	**0.2794**
SIN	H@5	0.1296	0.1608	0.2041	0.2650	0.4335	**0.4462**	*0.4446*	0.2832	0.2508	0.2785	*0.3148*	**0.3703**
	H@10	0.1933	0.1862	0.2784	0.3085	*0.5222*	**0.5348**	0.5174	0.4240	0.2863	0.4335	*0.4399*	**0.4715**
	N@5	0.1027	0.1169	0.1315	0.1588	0.3445	*0.3531*	**0.3534**	0.1957	0.1455	0.1924	*0.2231*	**0.2814**
	N@10	0.1476	0.1591	0.1556	0.1825	0.3728	**0.3815**	*0.3768*	0.2414	0.1713	0.2422	*0.2639*	**0.3141**

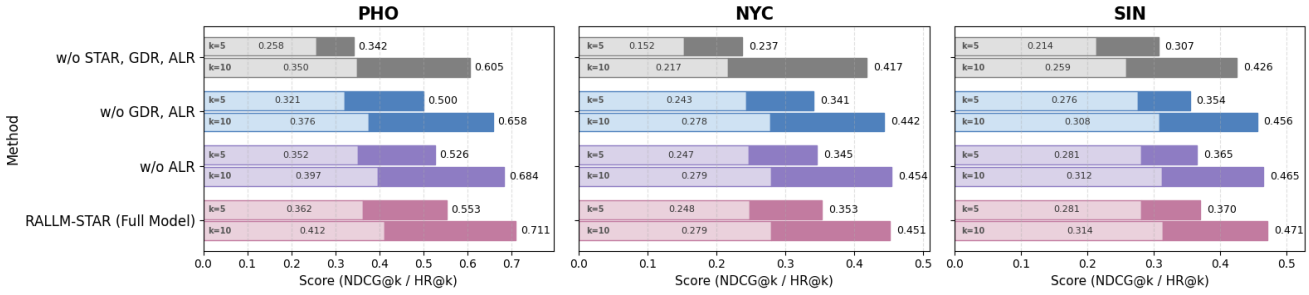


Figure 3: Ablation study of the proposed components STAR, GDR, and ALR in RALLM-STAR. For each bar, the light color represents NDCG score while the dark color represents the HR score. Results on PHO, NYC, and SIN are reported.

surpasses it by 0.2852, 0.1202, and 0.1630 in HR@10 on the 3 datasets, respectively. When compared to the graph-based methods, RALLM-STAR competes closely without any per-dataset training or hyperparameter tuning. These results demonstrate that spatial-temporal retrieval alignment can effectively substitute for learned graph representations in the zero-shot setting.

Comparison with zero-shot LLM baselines. Across all datasets, RALLM-STAR achieves substantial improvements over the zero-shot LLM baselines LLMMob, LLMMove, and LLMZS. For example, compared to LLMZS, our method surpasses it by 0.2105 in HR@5 on PHO and 0.0956 in NDCG@10 on NYC. These baselines rely primarily on prompt engineering and direct utilization of a user's recent check-ins, which often suffer from limited or noisy context. Without explicit retrieval guidance, LLMs may incorporate irrelevant movement patterns or hallucinate plausible but spatially invalid POIs. In contrast, RALLM-STAR leverages the STAR retriever to supply high-quality, behaviorally aligned mobility trajectories and the GDR reranker to impose path-level spatial realism. This grounded context dramatically improves the LLM's decision-making, yielding a remarkable HR@5 of 0.5526 and 0.3529 on PHO and NYC datasets.

Comparison with RALLM-POI. Compared with our earlier RALLM-POI framework [16], RALLM-STAR achieves consistent gains across all datasets, with average improvements of 3.30%, 31.05%, and 19.02% in NDCG@10 on PHO, NYC, and SIN respectively. The improvements are particularly pronounced on datasets with large POI vocabularies such as NYC. These gains validate the contribution of the two additional components introduced in our improved version: (1) The consideration of both long-term and short-term user historical interactions that enables more comprehensive query information, and (2) the probabilistic GMM-based grounding that increases retrieval precision under complex spatial distributions. Together, these additions significantly improve the reliability, robustness, and interpretability of the final recommendations, demonstrating meaningful advances over the earlier RALLM-POI model.

Dataset-level observations. RALLM-STAR delivers especially significant improvements on NYC and SIN, where the larger POI vocabularies and denser mobility flows make geographical grounding crucial. In PHO with more sparse and less POIs, the model maintains substantial advantages in both HR and

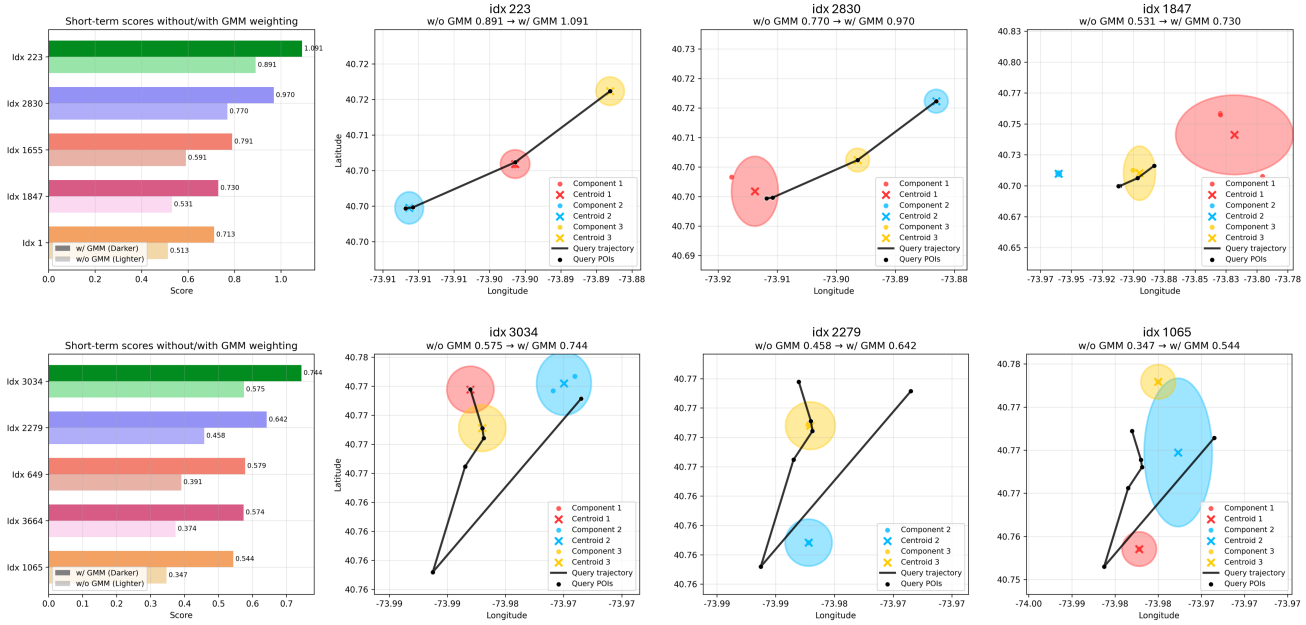


Figure 4: Visualization of the GMM effect in STAR on the NYC dataset. Top and bottom rows refer to two test user cases. For *each row*, the first bar plot displays the scores of the top-5 retrieved trajectories without and with GMM weighting, where darker bars refer to with GMM and lighter bars refer to without GMM. The subsequent three plots illustrate the user query trajectory alongside three candidate clusters.

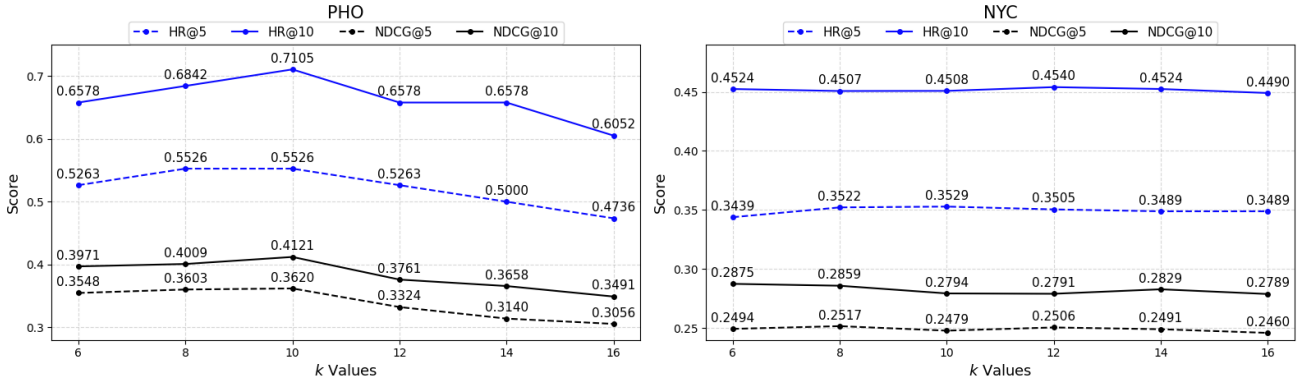


Figure 5: Performance comparison with different number of retrieved histories k.

NDCG, demonstrating robustness across urban structures of varying scale and complexity. Overall, the results confirm that combining retrieval-augmented modeling with spatial alignment and agentic correction enables RALLM-STAR to consistently outperform existing zero-shot LLM-based approaches while remaining competitive with recent supervised graph/hypergraph recommenders, without any per-dataset training.

4.3. Ablation Study

4.3.1. Analysis on the proposed components

We conduct an ablation study to evaluate the contribution of each major component in RALLM-STAR, as shown in Fig. 3. On all datasets, the full model (RALLM-STAR) achieves the best performance in both HR@5 and NDCG@5, demonstrating the effectiveness of our overall design. Removing the ALR module ('w/o ALR') results in consistent performance drops, indicating its importance for rectifying the responses of the LLM. Further removing GDR ('w/o GDR, ALR') further reduces performance. On PHO, HR@5 decreases from 0.5526 to 0.5000 and NDCG@5 from 0.3620 to 0.3210. This is because GDR serves as an essential module to rerank the retrieved context, enabling the LLM to better leverage the historical geographical information. Lastly, when all three components are removed ('w/o STAR, GDR, ALR'), the model's performance drops most significantly. Using NYC as an example, HR@5 is reduced to 0.2370, and NDCG@5 to 0.1521. Therefore, each proposed component contributes independently and cumulatively to the overall recommendation performance of RALLM-STAR. The synergistic effect of these

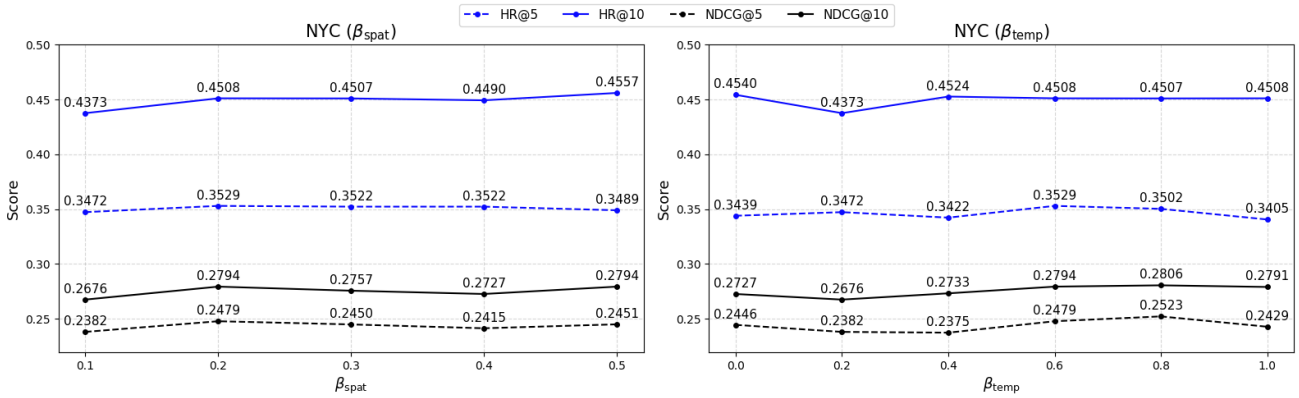


Figure 6: Performance comparison with different β_{spat} and β_{temp} values in STAR.

components demonstrates that combining STAR's enhanced retrieval mechanism with GDR's geographical reranking and ALR's LLM alignment creates a more robust and effective recommendation framework. This result validates our design philosophy of tightly integrating retrieval, reranking, and response refinement as complementary mechanisms.

4.3.2. Analysis of Retrieval and LLM Reasoning

To further understand the respective roles of retrieval and LLM reasoning in RALLM-STAR, we conduct a controlled analysis using GPT-4o-mini as the fixed backbone. We independently vary two factors: the use of STAR+GDR retrieval and the use of explicit natural-language reasoning. Specifically, the "Direct" variants remove the explicit reasoning process from the prompt and instruct the LLM to directly output ranked POI recommendations without explanations, rationales, or intermediate reasoning. The corresponding "Reason" variants retain the original reasoning prompt. Within each matched comparison, all other experimental settings are kept unchanged. In particular, RALLM-STAR (Direct) and RALLM-STAR (Reason) use the same retrieved and reranked trajectory context, so that their difference reflects the effect of explicit LLM reasoning.

We evaluate five configurations: *LLM-only (Direct)*, *LLM-only (Reason)*, *RALLM-STAR (Direct)*, *RALLM-STAR (Reason)*, and the full RALLM-STAR. Here, "Reason" denotes explicit natural-language reasoning. The results are reported in Table 3. The results reveal a clear difference between the contributions of retrieval and explicit reasoning. When explicit reasoning is disabled, introducing STAR and GDR substantially improves recommendation performance on all three datasets. HR@5 increases from 0.3421 to 0.5250 on PHO, from 0.2304 to 0.3406 on NYC, and from 0.3054 to 0.3627 on SIN. NDCG@5 shows the same trend, increasing from 0.2570 to 0.3508 on PHO, from 0.1508 to 0.2422 on NYC, and from 0.2107 to 0.2746 on SIN. Similar gains are observed when explicit reasoning is enabled, indicating that the benefit of STAR and GDR is consistently preserved under different reasoning settings. In comparison, enabling explicit reasoning while keeping the retrieval setting fixed produces smaller but consistent improvements. With STAR and GDR enabled, HR@5 increases from 0.5250 to 0.5263 on PHO, from 0.3406 to 0.3458 on NYC, and from 0.3627 to 0.3650 on SIN. NDCG@5 also improves from 0.3508 to 0.3522, from 0.2422 to 0.2471, and from 0.2746 to 0.2788, respectively. A similarly modest effect is observed when retrieval is disabled, where explicit reasoning only slightly improves the recommendation results.

Overall, these controlled comparisons indicate that the main performance gains of RALLM-STAR originate from the spatial-temporal retrieval and alignment provided by STAR and GDR, while explicit LLM reasoning offers an additional complementary benefit. The full model further improves the results through ALR-based response rectification, consistent with the component ablation analysis.

4.3.3. Visualization of the GMM effect in STAR

To provide empirical intuition for the spatial alignment mechanism in Equation (9), we visualize the effect of GMM-based spatial score re-calibration on two real-world test trajectories (Fig. 4). The visualization comprises three components: score redistribution, spatial composition of the retrieved candidates, and geometric alignment between query and GMM centroids. The left panel shows cosine similarity scores before and after GMM weighting. Trajectories with high semantic similarity (lighter bars) may not be spatially aligned with the query's recent mobility. The GMM reweighting (darker bars) recalibrates scores via $s'_{i,S} = s^{\text{temp}}_{i,S} + \beta_{\text{spat}}\, \hat{s}^{\text{spat}}_{i,S}$, promoting geographically coherent candidates. For instance, in the top scenario, the top candidate's score increases from 0.891 to 1.091, while spatially less-aligned candidates are increased by smaller margins. The center and right panels visualize candidate trajectories as fitted GMMs with $K = 3$

Table 3
Controlled analysis of spatial-temporal retrieval and explicit LLM reasoning.

Variant	Retrieval	Reason	ALR	PHO		NYC		SIN	
				HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5
LLM-only (Direct)	×	×	×	0.3421	0.2570	0.2304	0.1508	0.3054	0.2107
LLM-only (Reason)	×	✓	×	0.3421	0.2585	0.2370	0.1521	0.3072	0.2138
RALLM-STAR (Direct)	✓	×	×	0.5250	0.3508	0.3406	0.2422	0.3627	0.2746
RALLM-STAR (Reason)	✓	✓	×	0.5263	0.3522	0.3458	0.2471	0.3650	0.2788
Full RALLM-STAR	✓	✓	✓	**0.5526**	**0.3620**	**0.3529**	**0.2479**	**0.3703**	**0.2814**

components (shown as translucent Gaussian ellipses). Query trajectory POIs are marked as black dots. High-quality matches exhibit clear geometric overlap with query coordinates, while misaligned candidates show spatial components distant from the query, resulting in low spatial scores $\hat{g}_i$. This visualization validates our probabilistic approach: by computing expected log-density in Eq. 9, we naturally handle varied spatial distributions and ensure that retrieved trajectories are both semantically relevant and geographically plausible.

4.3.4. Analysis on the number of retrieved history k

We investigate the effect of varying the number of retrieved histories on recommendation performance, as shown in Fig. 5. For both the PHO and NYC datasets, we evaluate HR@5, HR@10, NDCG@5, and NDCG@10 as the number of retrieved history items k increases from 6 to 16. Across all metrics and datasets, performance peaks near k=10, where our enhanced STAR retriever achieves optimal balance. On PHO, HR@10 reaches 0.7105 at k=10, while on NYC the HR@10 remains stable around 0.45 across the k range, indicating robust retrieval quality. When k exceeds 10, we observe either stagnation or slight performance decline, indicating diminishing returns and possible noise introduction from less relevant histories. This improvement over prior ablations underscores the effectiveness of STAR's dual-stream temporal modeling and GMM-based spatial alignment in identifying semantically and spatially coherent historical trajectories. Based on these empirical results, we select k=10 as the optimal number of retrieved histories, providing a strong balance between contextual richness and retrieval precision. Although the NYC dataset exhibits comparable performance across $k \in [8, 12]$, we adopt the value 10 uniformly across all datasets to ensure methodological consistency and generalization, which serves as an effective and robust choice for providing sufficient context while avoiding noise.

4.3.5. Analysis on the effect of β_{spat} and β_{temp} in STAR

The hyperparameters β_{spat} and β_{temp} play crucial roles in balancing the information integrated by STAR. As depicted in the left panel of Fig. 6, we present the impact of β_{spat}. This hyperparameter controls the strength of additional spatial alignment guided by the GMM. A β_{spat} value of 0.2 yields the best performance. This indicates that the introduced additional spatial context serves as an effective refinement during retrieval. While incorporating spatial information ($\beta_{spat} > 0$) consistently improves performance over solely relying on temporal streams, excessively emphasizing it can lead to degradation in some metrics, as overly strict spatial constraints may filter out semantically relevant but geographically slightly distant recommendations. Thus we set β_{spat} to 0.2 for all datasets. On the other hand, the right panel of Fig. 6 illustrates β_{temp}, which governs the relative contributions of short-term and long-term behavioral patterns during the retrieval results fusion. When $\beta_{temp} = 0.6$, STAR achieves its peak performance on HR@5 of 0.3529, while setting $\beta_{temp} = 0.8$ gives the highest NDCG@5 of 0.2523. This suggests that jointly modeling short- and long-term user behaviors, with a slightly stronger emphasis on recent behavior, is more effective than relying on a single dominant trajectory. We set the value of β_{temp} to 0.6 for all datasets.

4.3.6. Analysis on the spatial modeling methods in STAR

To validate the effectiveness of our Gaussian Mixture Model (GMM) approach for spatial alignment in the STAR retrieval component, we conduct a comprehensive ablation study comparing GMM against several alternative spatial modeling techniques. As shown in Table 4, GMM demonstrates superior performance across both datasets, achieving the highest HR@10 of 0.7105 and NDCG@10 of 0.4121 on the PHO dataset, and HR@10 of 0.4508 and NDCG@10 of 0.2794 on the NYC dataset. K-Means clustering, which partitions spatial locations into spherical clusters, shows competitive performance with HR@10 of 0.6842 and 0.4440 on PHO and NYC respectively, but fails to capture the multimodal nature of spatial distributions common in urban mobility patterns. KD-Tree spatial indexing, designed for efficient nearest neighbor search, achieves HR@10 of 0.6842 and 0.4490 on PHO and NYC, demonstrating good spatial locality preservation but limited handling of irregular spatial relationships.

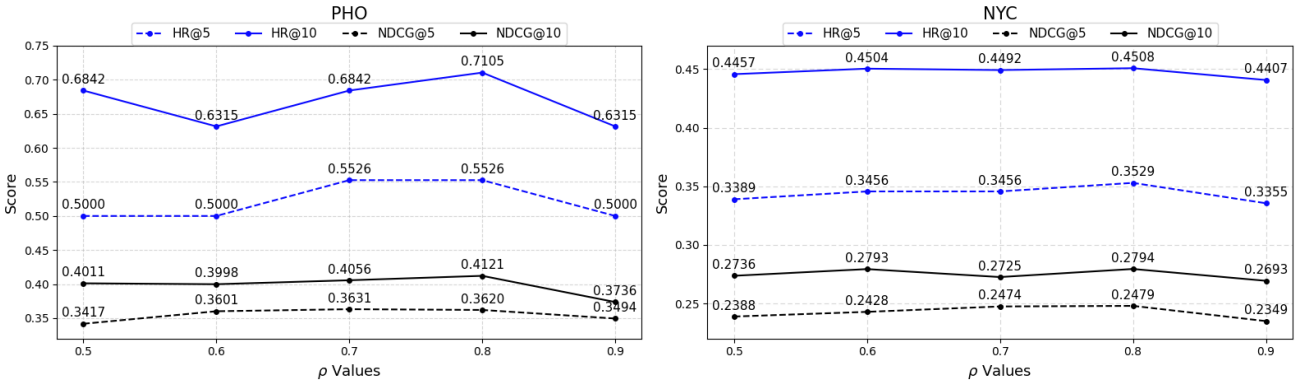


Figure 7: Performance comparison with different decaying weight ρ in DWDTW.

Table 4
Comparison of Different Spatial Modeling Methods

Method	PHO		NYC	
	HR@10	NDCG@10	HR@10	NDCG@10
GMM	0.7105	0.4121	0.4508	0.2794
KMeans	0.6842	0.3920	0.4440	0.2742
KNN	0.6578	0.3702	0.4407	0.2755
KDTree	0.6842	0.3871	0.4490	0.2780
DBSCAN	0.6578	0.3934	0.4507	0.2800

Table 5
Comparison of different long-term and short-term retrieval results fusion strategies in STAR.

Method	PHO		NYC	
	HR@10	NDCG@10	HR@10	NDCG@10
RRF	0.7105	0.4121	0.4508	0.2794
GeoMean	0.6315	0.3799	0.4490	0.2801
Linear	0.6578	0.3911	0.4502	0.2780
Borda	0.6842	0.3889	0.4457	0.2757
Interleave	0.6578	0.3875	0.4424	0.2690

K-Nearest Neighbors (KNN) spatial modeling, which relies on proximity-based density estimation, shows the lowest performance among non-GMM methods with HR@10 of 0.6578 and 0.4407 on PHO and NYC, indicating that simple distance-based approaches cannot effectively capture complex spatial mobility patterns. DBSCAN clustering, which identifies dense spatial regions and assigns noise points, performs competitively with HR@10 of 0.6578 and 0.4507 on PHO and NYC, showing moderate effectiveness in handling irregular spatial shapes but struggling with optimal parameter selection for variable-density urban environments. These findings validate our choice of GMM as the spatial modeling approach in STAR, demonstrating that probabilistic modeling with flexible covariance structures is essential for capturing the complex, multimodal spatial distributions inherent in human mobility patterns.

4.3.7. Analysis on the effect of long-term and short-term fusion strategies in STAR

The choice of fusion strategy significantly impacts the effectiveness of combining short-term and long-term retrieval results in STAR. Therefore, we compare RRF with (1) Geomean, which multiplies the normalized scores and raises them to the corresponding β_{temp} and $1-\beta_{temp}$ exponents to emphasize mutual agreement; (2) Linear fusion, which interpolates min–max normalized similarity scores from both streams using the same β_{temp} weighting as STAR; (3) Borda Count, which converts each ranked list into positional votes and sums the ballots with short/long weights; and (4) Interleave, which alternates items from the two ranked lists according to the dominant stream while keeping duplicates out. As shown in Table 5, RRF demonstrates superior performance across both datasets, achieving the highest HR@10 of 0.7105 and NDCG@10 of 0.4121 on the PHO dataset, and HR@10 of 0.4508 and NDCG@10 of 0.2794 on the NYC dataset. This confirms that rank-based reciprocal scoring effectively handles the heterogeneous distributions typical in dual-stream retrieval systems. Borda Count follows as the second-best performer with HR@10 of 0.6842 and 0.4457 on PHO and NYC respectively, validating the theoretical advantages of rank-based fusion over score-based approaches. Linear fusion shows competitive performance but slightly underperforms RRF, suggesting that simple score averaging may over-weight items with artificially inflated scores from either retrieval stream. The Geomean strategy exhibits the poorest performance, with HR@10 dropping to 0.6315 and 0.4490 on PHO and NYC, indicating that logarithmic operations can amplify score biases and reduce ranking stability. The Interleave method, while computationally efficient, fails to leverage score information effectively, resulting in suboptimal performance. These findings validate our choice of RRF as the fusion mechanism in STAR, demonstrating that

Table 6
The performance of different user groups across three datasets for user cold-start analysis.

User	PHO		NYC		SIN	
Groups	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5
Inactive	0.6154	0.4061	0.3571	0.2705	0.3529	0.2628
Normal	0.6923	0.5379	0.3556	0.2402	0.3947	0.3032
Very Active	0.3846	0.3178	0.2762	0.1975	0.4434	0.3371

information retrieval theory provides more reliable strategies for combining heterogeneous retrieval results than ad-hoc mathematical operations.

4.3.8. Analysis on the decaying weight ρ in DWDTW

To understand the contribution of spatial reranking, we ablate the the hyperparameter ρ in DWDTW, which adjusts the spatial alignment cost. Higher ρ values will raise the importance of more recent locations in user trajectories, thereby influencing the spatial consistency of the reranked candidate trajectories. The results of adjusting ρ is shown in Fig. 7. For the PHO dataset, increasing ρ from 0.6 to 0.8 improves HR@10 from 0.6315 to 0.7105 and NDCG@10 from 0.3998 to 0.4121, indicating that giving stronger priority to recent trajectory points enhances the geographical plausibility of the retrieved trajectories. However, further increasing ρ to 0.9 results in a large decline in performance on both PHO and NYC datasets, suggesting that excessive emphasis on recent points could potentially overlook contextually relevant diversity to the retrieved examples. Similarly, for the NYC dataset, HR@5 experiences a peak improvement at 0.8. Therefore, we set $\rho = 0.8$ as a point of balance, achieving robust recommendation performance improvements while maintaining sufficient spatial breadth in candidate reranking. These results reinforce the notion that an optimal range of ρ balances the recency bias effectively, allowing the reranker to retain fine-grained spatial alignment without overfitting to the immediate past at the expense of wider trajectory comprehension.

4.3.9. User cold-start performance

The cold-start problem in recommendation systems presents a significant challenge, with active users tending to provide more historical data and generally easier to learn, while inactive users provide limited historical trajectory and are hard to capture their visit patterns [19]. We divide users into three tiers, including Inactive, Normal, and Very Active, based on their historical POI counts. These groups correspond to the bottom 35%, middle 30%, and top 35% of the user base, respectively. This categorization allows us to evaluate the ability of our model to adapt to varying levels of user activity and historical data availability. As shown in Table 6, our analysis across three datasets, namely PHO, NYC, and SIN, indicates that normal and inactive users often achieve higher performance metrics despite providing less historical data. For instance, inactive users in PHO achieve an NDCG@5 of 0.4061, outperforming very active users. Similarly, in NYC, inactive users achieve an HR@5 of 0.3571 and NDCG@5 of 0.2705, which is much higher compared to very active users, who achieve only 0.2762 and 0.1975 respectively. A similar trend is observed in SIN where normal users achieve stronger performance relative to other groups. This is attributed to our retrieval based approach STAR. By identifying and incorporating trajectories from other users who have exhibited similar behavioral patterns, our model could compensate for the lack of sufficient historical data in inactive users, thereby enriching the feature space for cold-start scenarios.

Table 7
Comparison with RAG retrieval strategies.

Method	PHO		NYC		SIN	
	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5
GraphRAG	0.4474	0.2791	0.3255	0.2196	0.3671	0.2778
Speculative RAG	0.5000	0.3158	0.3439	0.2434	0.3608	0.2786
Corrective RAG	0.5000	0.3474	0.3389	0.2458	0.3671	0.2847
RALLM-STAR	**0.5526**	**0.3620**	**0.3529**	**0.2479**	**0.3703**	**0.2814**

4.3.10. Analysis on the RAG strategy

To provide a more comprehensive comparison of retrieval strategies, we further include three modern RAG frameworks as baselines: GraphRAG [4], which constructs a knowledge graph over the corpus for retrieval-augmented reasoning; Speculative RAG [37], which uses a speculative decoding-inspired retrieval pipeline; and Corrective RAG [39], which incorporates a self-corrective document assessment module. All four RAG methods (including RALLM-STAR) use the same LLM for fair comparison. As shown in Table 7, RALLM-STAR achieves the best HR@5 and NDCG@5 on all three datasets (e.g., PHO HR@5 0.5526 vs. 0.5000 for the best compared RAG approach), confirming that generic RAG strategies alone are insufficient for the POI domain and that spatial-temporal retrieval design provides unique benefits.

Table 8
Prompt-order sensitivity test for GDR.

Order	PHO		NYC		SIN	
	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5
DWDTW	**0.5526**	**0.3620**	**0.3529**	**0.2479**	**0.3703**	**0.2814**
Reverse	0.4737	0.3363	0.3239	0.2323	0.3622	0.2803
Random	0.5263	0.3608	0.3339	0.2371	0.3671	0.2806

4.3.11. Prompt-Order Sensitivity of GDR

To validate that the DWDTW ordering within GDR directly benefits LLM performance, we conduct a prompt-order sensitivity test. We keep the same three GDR-selected exemplars and compare three prompt arrangements: the original DWDTW order, reverse order of DWDTW, and random order. The results in Table 8 confirm that the DWDTW order consistently achieves the best HR@5 and NDCG@5 across all three datasets. The reverse order consistently underperforms, demonstrating that GDR is not merely a pre-filter, it also determines the priority ordering within the prompt, and this ordering exerts a measurable and consistent effect on LLM performance under a fixed token budget.

4.4. Large-scale Evaluation

To assess RALLM-STAR under a substantially larger user and POI space, we conduct experiments on the Gowalla dataset. As shown in Table 9, RALLM-STAR remains among the strongest zero-shot methods: it achieves the best HR@5, NDCG@5, and NDCG@10 among zero-shot methods, and the second-best zero-shot HR@10. It also remains competitive with recent training-based graph/hypergraph baselines despite requiring no dataset-specific training. This is important for scalability: on a large dataset such as Gowalla, RALLM-STAR avoids multi-epoch neural training and large GPU-memory requirements, while training-based baselines must rebuild and optimize dataset-specific sequential or graph models. These results suggest that the proposed retrieval-and-reranking pipeline continues to provide strong ranking quality when both the user population and POI candidate space become much larger.

4.5. Cross-Backbone Robustness

To evaluate the robustness of RALLM-STAR across different LLM backbones, we apply the identical pipeline (STAR + GDR + ALR with the same hyperparameters) on the PHO, NYC, and SIN test sets using six LLMs: GPT-4o-mini, Grok-4.1-Fast, GLM-4.7-Flash, Qwen2.5-72B, DeepSeek-V3.2, and Qwen3-30B-A3B. The selected backbones cover models with different architectures, scales, and reasoning characteristics, allowing us to examine whether the effectiveness of RALLM-STAR depends on a particular underlying LLM.

Table 9
Large-scale evaluation on the Gowalla dataset. Within each method group, best results are highlighted in bold and second-best results are shown in italics.

Method	HR@5	NDCG@5	HR@10	NDCG@10
Training-based Methods				
ALSTM	0.1720	0.1445	0.1770	0.1462
MCRNN	0.1860	0.1519	0.1980	0.1558
CTLE	0.1810	0.1518	0.1920	0.1553
CFPRec	0.1690	0.1399	0.1800	0.1434
DCHL	0.1936	*0.1535*	0.2132	0.1588
CAMSHG	*0.2095*	0.1526	*0.2291*	*0.1593*
MSAHG	**0.2215**	**0.1611**	**0.2830**	**0.1826**
Zero-shot Methods				
LLMMob	*0.2100*	0.1524	0.2670	0.1701
LLMMove	0.1860	0.1246	0.2550	0.1471
LLMZS	*0.2100*	*0.1552*	0.2740	0.1757
RALLM-POI	0.2030	0.1499	**0.2880**	*0.1772*
RALLM-STAR	**0.2120**	**0.1567**	*0.2790*	**0.1794**

Table 10
Cross-backbone evaluation of RALLM-STAR across six LLMs on three datasets.

Backbone	PHO		NYC		SIN	
	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5
GPT-4o-mini	**0.5526**	**0.3620**	0.3529	0.2479	0.3703	0.2814
Grok-4.1-Fast	0.5263	0.3581	**0.3957**	**0.2870**	**0.4256**	**0.3407**
GLM-4.7-Flash	0.5000	0.3524	0.3422	0.2521	0.3551	0.2602
Qwen2.5-72B	0.4474	0.3423	0.2888	0.2024	0.3402	0.2566
DeepSeek-V3.2	0.5263	0.3576	0.3439	0.2430	0.3582	0.2706
Qwen3-30B-A3B	0.5000	0.3572	0.3508	0.2455	0.3691	0.2760

As shown in Table 10, RALLM-STAR achieves stable recommendation performance across all six backbones and three datasets. GPT-4o-mini performs best on PHO, whereas Grok-4.1-Fast achieves the strongest results on NYC and SIN. The two reasoning-oriented backbones, DeepSeek-V3.2 and Qwen3-30B-A3B, also maintain competitive performance across all datasets. In particular, Qwen3-30B-A3B achieves results close to GPT-4o-mini on NYC and SIN despite their different model characteristics. Overall, the results demonstrate that RALLM-STAR is robust to the choice of the underlying LLM and remains effective across both general-purpose and reasoning-oriented backbones.

4.6. Computational Cost Analysis

We profile the component-level computational cost of RALLM-STAR across all three datasets, as summarized in Table 11. The local STAR/GDR operations are lightweight after initialization: retrieval remains below 0.5 s/query and DWDTW reranking below 0.015 s/query on all datasets. GMM fitting is a one-time initialization cost (0.9-14 s depending on dataset size). The dominant costs are the two LLM-facing blocks (GDR LLM call and ALR second LLM call), each requiring 4-8 s per query. The total prompt token

Table 11
Component-level computational cost of RALLM-STAR.

Module	Block	PHO		NYC		SIN	
		Time	API Cost	Time	API Cost	Time	API Cost
One-time Initialization							
STAR	GMM fitting	0.913 s	–	13.844 s	–	14.143 s	–
Per-query Cost							
STAR	Retrieval	0.003 s	–	0.438 s	–	0.311 s	–
GDR	DWDTW	0.001 s	–	0.012 s	–	0.014 s	–
GDR	LLM call	5.644 s	583.3 tokens	5.850 s	840.7 tokens	4.143 s	562.3 tokens
ALR	2nd LLM call	5.860 s	816.7 tokens	5.421 s	1075.0 tokens	8.188 s	790.7 tokens

consumption per query remains modest, approximately 1,400-1,900 tokens across all datasets. These results demonstrate that the proposed method is computationally efficient, with lightweight local operations and low API overhead, while ALR explicitly trades one additional API call for improved output robustness.

4.7. Case Study

Case 1: RALLM-STAR Refers to Retrieved Trajectories for Hard Case

Long-term Trajectory: [5117, 11723, 11405, 11358, 13609, 13510, 7736, 12010, 11358, 12479, 15097, 979, 15154, 14909, 11723, 9906, 12690, 12950, 13920, 12053, 10961, 6619, 4734, 6892, 14343, 6975, 4319, 12897, 13510, 11405, 15007, 11358, 11520, 3162, 9011, 7799, 8116, 15154, 14909, ..., 15107]
Short-term Trajectory: [4319, 11467, 12897, 13510, 11405, 15007, 5062, 15107, 14654]
Retrieved Trajectories: [[13097, 14343, 4182, 5061, 4319, 1647, 13510, 12950, 11358, 13510, 13840, 12053, 3780, 11462, 13616, 14909], ..., [12479, 13609, 12950, 13510, 9906, 14654, 14909, 15356], ..., [8116, 14654, 14909, 4319, 12010, 6892, 13609], [4182, 12897, 5061, 2117, 4319], [12479, 13510, 12950, 6829, 5062, 4054]].

Response: "recommendation": [13510, 11405, 12897, 15154, 14909, 12053, 11358, 4319, 15007, 13609],
"reason": The user's long-term check-ins indicate a strong preference for POIs 13510, 11405, and 12897, which are frequently revisited and. Recent check-ins also show interest in 15154 and 14909, which are also popular among candidate trajectories. The recommendation includes popular POIs while also considering potential new interests based on context visit transitions and recent user behavior. The additional POIs are included to ensure diversity and to explore other categories that may align with the user's preferences.
Ground Truth: 14909.

We show and analyze some challenging cases from the NYC dataset for better understanding of the capabilities of RALLM-STAR. In Case Study 1, the retrieved information with STAR proves effective by leveraging the user's trajectories along with retrieved

trajectories. Specially, The model successfully prioritizes POI 14909, which is not only present in the user's historical check-ins but was the last POI in the top-1 retrieved trajectory, aligning perfectly with the ground truth. The result demonstrates the effectiveness of providing useful contextual trajectories for LLMs as well as prioritizing geographically nearby locations.

In Case Study 2, the ALR refines initial recommendations by accurately prioritizing the user's long-term preference for the ground truth POI 7026, which the prior LLM missed. The case demonstrates its importance in correcting and enhancing base model predictions. These cases showcase the strengths of the proposed modules in improving recommendation accuracy and personalization.

Case 2: RALLM-STAR Rectifies its Recommendation List with ALR

Long-term Trajectory: [3214, 7026, 4279, 11326, 3006, 6574, 4279, 7026, 13028, 11940, 11940, 4279, 14011, 13446, 6360, 4684, 1768, 4279]
Short-term Trajectory: [6360, 4684, 1768, 4279]
Recommendation from the prior LLM: [4279, 6360, 4684, 1768, 3214, 14011, 11326, 13028, 11940, 13446]
Recommendation of the rectifier LLM: [3214, 14011, 11326, 13028, 11940, 7026, 6574, 1768, 6360, 4684]
Ground Truth: 7026.

5. Conclusion

This paper introduces RALLM-STAR, a novel framework for zero-shot next POI recommendation that effectively combines the generalization capabilities of LLMs with retrieval augmented generation and self-rectification. Unlike traditional training-based methods that require significant resources for retraining and tuning, RALLM-STAR leverages inherent knowledge within LLMs, augmented by contextually and geographically relevant historical trajectories with STAR and refined by GDR. The ALR further enforces quality assurance by equipping the LLM with self-evaluation and correction capabilities. Through comprehensive evaluations on three real-world datasets, plus large-scale validation on Gowalla, RALLM-STAR demonstrates significant performance gains over state-of-the-art zero-shot LLM methods and competes closely with recent graph/hypergraph recommenders without any per-dataset training. The ablation and analysis reveal how each component synergistically contributes to accuracy and robustness, particularly excelling in addressing the cold-start problem for inactive users while maintaining strong adaptability for diverse user trajectories. Component-level profiling confirms that the pipeline is lightweight, and the cross-backbone experiments further verify the generalization ability of the RALLM-STAR.

Acknowledgments

This research is supported by the Ministry of Education, Singapore (MOE), under its Academic Research Fund Tier 2 (Award No. MOE-T2EP20123- 0015). Any opinions, findings and conclusions, or recommendations expressed in this material are those of the authors and do not reflect the views of MOE.

CRediT authorship contribution statement

Kunrong Li: Writing – review & editing, Writing – original draft, Visualization, Validation, Investigation, Formal analysis, Data curation, Conceptualization. **Kwan Hui Lim:** Supervision, Writing – review & editing, Investigation, Project administration, Funding acquisition.

References

[1] Beneduce, C., Lepri, B., Luca, M., 2025. Large language models are zero-shot next location predictors. IEEE Access .

[2] Cormack, G.V., Clarke, C.L., Buettcher, S., 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods, in: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, pp. 758–759.

[3] Duan, C., Fan, W., Zhou, W., Liu, H., Wen, J., 2023. Clsprec: Contrastive learning of long and short-term preferences for next poi recommendation, in: CIKM, pp. 473–482.

[4] Edge, D., Trinh, H., Cheng, N., Bradley, J., Chao, A., Mody, A., Truitt, S., Metropolitansky, D., Ness, R.O., Larson, J., 2024. From local to global: A graph rag approach to query-focused summarization. arXiv preprint arXiv:2404.16130 .

[5] Feng, S., Lyu, H., Li, F., Sun, Z., Chen, C., 2024. Where to move next: Zero-shot generalization of llms for next poi recommendation, in: CAI.

[6] Guan, S., Xiong, H., Wang, J., Bian, J., Zhu, B., Lou, J.g., 2025. Evaluating llm-based agents for multi-turn conversations: A survey. arXiv preprint arXiv:2503.22458 .

[7] Halder, S., Lim, K.H., Chan, J., Zhang, X., 2021. Transformer-based multi-task learning for queuing time aware next poi recommendation, in: Pacific-Asia conference on knowledge discovery and data mining, Springer. pp. 510–523.

[8] Halder, S., Lim, K.H., Chan, J., Zhang, X., 2023. Capacity-aware fair poi recommendation combining transformer neural networks and resource allocation policy. Applied Soft Computing 147, 110720.

[9] Harte, J., Zorgdrager, W., Louridas, P., Katsifodimos, A., Jannach, D., Fragkoulis, M., 2023. Leveraging large language models for sequential recommendation, in: Proceedings of the 17th ACM Conference on Recommender Systems, pp. 1096–1102.

[10] Hossain, M.B., Arefin, M.S., Sarker, I.H., Kowsher, M., Dhar, P.K., Koshiba, T., 2022. Caran: A context-aware

recency-based attention network for point-of-interest recommendation. IEEE Access 10, 36299–36310.

[11] Huang, L., Ma, Y., Wang, S., Liu, Y., 2019. An attention-based spatiotemporal lstm network for next poi recommendation. ITSC 14, 1585–1597.

[12] Jayaram, V., Ramineni, V., Krishnappa, M.S., 2024. Mitigating order sensitivity in large language models for multiple-choice question tasks. IJAIRD .

[13] Jeong, Y.S., Jeong, M.K., Omitaomu, O.A., 2011. Weighted dynamic time warping for time series classification. Pattern recognition 44, 2231–2240.

[14] Kim, H., Mitra, K., Chen, R.L., Rahman, S., Zhang, D., 2024. Meganno+: A human-llm collaborative annotation system. arXiv preprint arXiv:2402.18050 .

[15] Lai, Y., Su, Y., Wei, L., He, T., Wang, H., Chen, G., Zha, D., Liu, Q., Wang, X., 2024. Disentangled contrastive hypergraph learning for next poi recommendation, in: Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval, pp. 1452–1462.

[16] Li, K., Lim, K.H., 2025. Rallm-poi: Retrieval-augmented llm for zero-shot next poi recommendation with geographical reranking. arXiv preprint arXiv:2509.17066 .

[17] Li, K., Liu, X., Chen, Z., 2024a. Semantic consistency regularization with large language models for semi-supervised sentiment analysis, in: International Conference on Neural Information Processing, Springer. pp. 289–303.

[18] Li, K., Sun, Z., Lim, K.H., 2025. Hymoerec: Hybrid mixture-of-experts for sequential recommendation. arXiv preprint arXiv:2511.06388 .

[19] Li, P., de Rijke, M., Xue, H., Ao, S., Song, Y., Salim, F.D., 2024b. Large language models for next point-of-interest recommendation, in: SIGIR, pp. 1463–1472.

[20] Liao, D., Liu, W., Zhong, Y., Li, J., Wang, G., 2018. Predicting activity and location with multi-task context aware recurrent neural network., in: IJCAI.

[21] Lim, K.H., Chan, J., Karunasekera, S., Leckie, C., 2019. Tour recommendation and trip planning using location-based social media: A survey. KAIS 60, 1247–1275.

[22] Lin, Y., Li, Y., Xing, J., Fan, Z., 2026. Multifaceted scenario-aware hypergraph learning for next poi recommendation, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), pp. 15269–15277.

[23] Lin, Y., Wan, H., Guo, S., Lin, Y., 2021. Pre-training context and time aware location embeddings from spatial-temporal trajectories for user next location prediction, in: AAAI, pp. 4241–4248.

[24] Lin, Z., Zhang, X., Zhao, K., Lu, X., Zhang, Y., 2025. Strajrag: Supervised trajectory retrieval augmented generation for next poi recommendation with travel semantics. Information Processing & Management 62, 104235.

[25] Luo, S., Xu, J., Zhang, X., Wang, L., Liu, S., Hou, H., Song, L., 2025. Rallrec+: Retrieval augmented large language model recommendation with reasoning. Expert Systems with Applications , 129508.

[26] Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., et al., 2023. Self-refine: Iterative refinement with self-feedback. Advances in neural information processing systems 36, 46534–46594.

[27] Müller, M., 2007. Dynamic time warping. Information retrieval for music and motion , 69–84.

[28] Pezeshkpour, P., Hruschka, E., 2023. Large language models sensitivity to the order of options in multiple-choice questions. arXiv preprint arXiv:2308.11483 .

[29] Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., Yao, S., 2023. Reflexion: Language agents with verbal reinforcement learning. Advances in neural information processing systems 36, 8634–8652.

[30] Singh, A., Ehtesham, A., Kumar, S., Khoei, T.T., 2025. Agentic retrieval-augmented generation: A survey on agentic rag. arXiv preprint arXiv:2501.09136 .

[31] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. NeurIPS 30.

[32] Wan, K., Gao, L., Mu, H., Nakov, P., Wang, Y., Chen, X., 2025. A fano-style accuracy upper bound for llm single-pass reasoning in multi-hop qa. arXiv preprint arXiv:2509.21199 .

[33] Wang, S., Fan, W., Feng, Y., Shanru, L., Ma, X., Wang, S., Yin, D., 2025a. Knowledge graph retrieval-augmented generation for llm-based recommendation, in: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 27152–27168.

[34] Wang, T., Lai, Y., Wang, Y., Xiang, J., 2024. Dyagl: A dynamic-aware adaptive graph learning network for next poi recommendation, in: Pacific Rim International Conference on Artificial Intelligence, Springer. pp. 361–374.

[35] Wang, X., Fang, M., Zeng, Z., Cheng, T., 2023. Where would i go next? large language models as human mobility predictors. arXiv preprint arXiv:2308.15197 .

[36] Wang, Z., Zhu, Y., Zhang, Q., Liu, H., Wang, C., Liu, T., 2022. Graph-enhanced spatial-temporal network for next poi recommendation. ACM Transactions on Knowledge Discovery from Data (TKDD) 16, 1–21.

[37] Wang, Z.R., Wang, Z., Le, L., Zheng, H.S., Mishra, S., Perot, V., Zhang, Y., Mattapalli, A., Taly, A., Shang, J., et al., 2025b. Speculative rag: Enhancing retrieval augmented generation through drafting, in: International Conference on Learning Representations, pp. 18483–18505.

[38] Wu, Y., Jiao, X., Hao, Q., Xiao, Y., Zheng, W., 2024. Dlan: Modeling user long-and short-term preferences based on double-layer attention network for next point-of-interest recommendation. Journal of Intelligent & Fuzzy Systems 46, 3307–3321.

[39] Yan, S.Q., Gu, J.C., Zhu, Y., Ling, Z.H., 2024. Corrective retrieval augmented generation. arXiv preprint arXiv:2401.15884 .

[40] Yang, D., Zhang, D., Qu, B., 2016. Participatory cultural mapping based on collective behavior data in location-based social networks. ACM Transactions on Intelligent Systems and Technology (TIST) 7, 1–23.

[41] Yang, Q., Hu, S., Zhang, W., Zhang, J., 2022a. Attention mechanism and adaptive convolution actuated fusion network for next poi recommendation. International Journal of Intelligent Systems 37, 7888–7908.

[42] Yang, S., Liu, J., Zhao, K., 2022b. 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, pp. 1144–1153.

[43] Yin, F., Liu, Y., Shen, Z., Chen, L., Shang, S., Han, P., 2023. Next poi recommendation with dynamic graph and explicit dependency, in: Proceedings of the AAAI conference on artificial intelligence, pp. 4827–4834.

[44] Zhang, L., Sun, Z., Wu, Z., Zhang, J., Ong, Y.S., Qu, X., 2022. Next point-of-interest recommendation with inferring multi-step future preferences., in: IJCAI.

[45] Zhang, Q., Yang, P., Yu, J., Wang, H., He, X., Yiu, S.M., Yin, H., 2025. A survey on point-of-interest recommendation: Models, architectures, and security. IEEE Transactions on Knowledge and Data Engineering .

[46] Zheng, Z., Chao, W., Qiu, Z., Zhu, H., Xiong, H., 2024. Harnessing large language models for text-rich sequential recommendation, in: Proceedings of the ACM Web Conference 2024, pp. 3207–3216.

[47] Zhong, C., Zhu, J., Xi, H., 2021. Ps-lstm: Popularity analysis and social network for point-of-interest recommendation in previously unvisited locations, in: Proceedings of the 2021 2nd International Conference on Computing, Networks and Internet of Things, pp. 1–6.

[48] Zhou, Q., Ye, J., Zhou, W., Wen, J., 2025. Multi-scale hypergraph representation learning with channel attention for next poi recommendation. Neurocomputing , 131908.