entropy

MDPI

Article

ASKG: Adaptive Spatio-Temporal Knowledge Grounding for Candidate-Constrained Point-of-Interest Recommendation

Ruojia Tong [1], Tianxiang He [2], Tianyu Li [1], Junlin Zhou [1,2,*], Yan Fu [1,2] and Duanbing Chen [1,2,3]

[1] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; rjtong@std.uestc.edu.cn (R.T.); litianyu1166@163.com (T.L.); fuyan@uestc.edu.cn (Y.F.); dbchen@uestc.edu.cn (D.C.)
[2] Chengdu Union Big Data Technology Incorporated, Chengdu 610041, China; hetianxiang@unionbigdata.com
[3] Suining Institute of Digital Economy, Suining 629018, China
* Correspondence: jlzhou@uestc.edu.cn

Abstract

Candidate-constrained point-of-interest (POI) recommendation asks a large language model (LLM) to infer a user's next destination from an ordered trajectory and a fixed set of candidate descriptions. The model expresses its decision by generating the corresponding candidate label. Unlike item recommendation, this task must reconcile semantic preference with physical reachability and changes in visit intent. However, existing retrieval-augmented methods often select evidence by text similarity and serialize graph facts as independent sentences. They can consequently retrieve a topically relevant but geographically implausible venue and lose the relations that explain how a candidate fits the user's current context. We propose Adaptive Spatio-Temporal Knowledge Grounding (ASKG) to carry visit-relevant POI evidence from retrieval to label generation. ASKG adjusts graph-semantic and geographic recall to the user's activity range, gives greater weight to evidence associated with recent visits, and preserves typed relations within a fixed graph-token budget. The candidate prompt then queries these graph tokens directly. A candidate-aligned objective trains the LLM to distinguish among the valid labels. It is a weighted combination of full-vocabulary language modeling and a cross-entropy loss restricted to the n valid candidate labels. Under a candidate-constrained protocol with 10 options on Yelp and 20 on Foursquare, the reported experiments with three LLM backbones per dataset show that ASKG attains the highest MRR in all six dataset–backbone settings and the highest ACC in five. On average across the six dataset–backbone settings, ASKG achieves a relative improvement of 5.9% in MRR and 8.3% in ACC over the strongest baseline.

Keywords: POI recommendation; large language models; retrieval-augmented generation; graph–language fusion; spatio-temporal modeling

Check for updates

Academic Editors: Deniz Gençağa and Rita Singh

Received: 17 September 2026
Revised: 1 November 2026
Accepted: 13 November 2026
Published: 16 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Recommendation systems help users navigate large and rapidly changing collections of items. Generative recommendation provides a distinct modeling paradigm in which a language model interprets user behavior, item descriptions, task instructions, and external knowledge through a shared language interface and directly produces a recommendation decision [1]. RecGPT explores domain-oriented generative pre-training [2], while LLaRA aligns recommendation representations with the language model rather than relying on textual prompting alone [3]. Given this generative framework, our study asks

how spatio-temporal and relational knowledge can be retrieved and grounded to improve candidate ranking.

POI recommendation is a particularly demanding instance of this paradigm and involves distinct modeling and architectural requirements [4]. Venue descriptions reveal semantic preference, but the next visit also depends on whether a destination is reachable and whether it matches the user's current intent. These constraints vary across users and over the course of a trajectory. POIs further participate in category, business-area, geographic, and service relations that surface descriptions do not reliably recover. A generative POI recommender must therefore connect semantic evidence with mobility and relational context before producing its decision. Physical travel and trajectory-dependent intent make this setting different from ordinary product or media recommendation.

A first difficulty arises from the mismatch between mobility signals and language-model representations. Classical POI recommenders encode trajectory regularities through recurrent memory, preference modules, geographic attention, or explicit spatio-temporal attention [5–8]; DSDRec combines trajectory semantics with global spatio-temporal area encoding in a diffusion-based model [9]. These methods operate directly on coordinates and timestamps, whereas an LLM processes discrete tokens. Raw numerical values can be placed in a prompt, but fluent reading alone does not ensure that token representations preserve distance, relation type, or recency well enough to affect candidate ranking. Effective grounding therefore requires an interface that carries continuous mobility structure into the LLM's token space.

A second difficulty is evidence selection. Existing LLM-based POI methods represent venue content, personal memory, urban context, and geographic cues in language-oriented forms [10–15]. Yet, a POI platform contains far more relations than a single request can use. The retriever must select facts that are relevant to the candidates while respecting the travel range and recent intent expressed by the trajectory. Text similarity addresses relevance but can still favor a distant venue or an interest that is no longer current. Retrieval must consequently align semantic evidence with the spatio-temporal conditions of the request.

Knowledge-enhanced and retrieval-augmented generation offer a natural route to address this gap. They can supply request-specific facts without repeatedly encoding the entire platform into the LLM. Textual knowledge prompting retrieves and verbalizes relevant facts [16]. Graph-based retrieval retains a task-relevant subgraph and exposes it through learned representations [17]. K-RagRec retrieves knowledge-graph structure for LLM recommendation [18], and KERAG-R selects external relational knowledge through a recommendation-oriented GraphRAG component [19]. However, these methods treat language interfacing, mobility constraints, and knowledge retrieval as separate concerns. Their retrieval objectives do not jointly enforce semantic relevance and spatio-temporal feasibility. Moreover, verbalizing triples as independent sentences discards the typed paths connecting historical POIs, attributes, and candidates before the evidence reaches the candidate-label decision. Directly transferring these methods to candidate-constrained POI recommendation therefore leaves both difficulties unresolved. A POI generator instead needs evidence selection, relation-preserving representation, and label-level competition to form one continuous pathway.

We address these two difficulties with ASKG, an Adaptive Spatio-Temporal Knowledge Grounding framework. User mobility defines the feasible search region, while recent visits indicate which interests are currently active. ASKG uses these signals to divide a fixed retrieval budget between graph-semantic and geographic recall and to re-rank the resulting evidence. The retained POIs are expanded into typed local subgraphs and compressed into a fixed number of graph tokens. History, evidence, and candidate tokens remain separate, allowing the candidate prompt to query each source through gated cross-attention.

A two-stage schedule introduces this graph pathway gradually, and a candidate-aligned objective connects the grounded representation to the option labels used at inference. This design forms a continuous path from request-specific retrieval to candidate ranking while preserving the relations selected for the current trajectory.

ASKG is positioned as the ranking component of a candidate-generation-then-ranking architecture, a separation used in large-scale recommendation [20]. An upstream component would first retrieve a manageable candidate set from the POI inventory; ASKG would then compare those candidates using their option-label logits. This system-level candidate generation is distinct from ASKG's internal retrieval and re-ranking of graph evidence. Our benchmark supplies controlled candidate sets containing the observed next POI rather than measuring an implemented upstream retriever. Consequently, the results establish candidate-constrained ranking performance conditional on target inclusion, not full-inventory recall, end-to-end large-scale recommendation, or open-vocabulary generation. The research question concerns knowledge grounding within an LLM-based recommender, not whether an LLM universally outperforms conventional recommendation models.

The contributions of this work are:

- We identify a representation-alignment challenge in grounding candidate-constrained POI recommendation. Continuous spatio-temporal and relational signals must be aligned with the discrete token space used by an LLM.
- We propose an adaptive spatio-temporal retrieval mechanism. It adjusts graph-semantic and geographic evidence according to the user's mobility range and uses trajectory position to represent recent intent.
- We connect the retrieved relations to candidate-label prediction through a relation-preserving graph–language interface. Typed subgraphs are compressed to a fixed token budget, queried by the candidate prompt, and trained with direct competition among the valid option labels.
- We evaluate candidate-constrained ranking on Yelp and Foursquare using three LLM backbones per dataset. Under the reported 10/20-option protocols, ASKG obtains the highest MRR in all six settings and the highest ACC in five, with average relative improvements of 5.9% and 8.3% over the strongest baseline for each metric. These findings concern the implemented LLM-based evidence pipelines, not full-inventory recall or cross-paradigm superiority.

2. Related Work

Prior work differs in how it supplies evidence. Language-based methods encode behavior and item content for a generator. Spatio-temporal methods derive mobility constraints from trajectories, whereas retrieval-based methods introduce external facts at inference time. These three roles are rarely combined in a unified framework. Some methods incorporate more than one role, but evidence selection, mobility awareness, and language generation usually remain loosely coupled. ASKG addresses this fragmentation through a jointly conditioned pipeline.

Language-conditioned generative recommendation. This family converts recommendation inputs into tokens or learned representations that a language model can interpret. P5 provides a shared text-to-text interface [1]. RecGPT improves recommendation-language understanding through domain pre-training [2]. However, these approaches do not explicitly align continuous behavioral signals with the discrete token space used by the language model. LLaRA adds collaborative signals that item text does not contain [3]. Later methods improve how recommendation targets are represented: GRAM fuses multi-granular item semantics [21], and GNPR-SID assigns semantic identifiers to POIs [22]. LLM4POI serializes trajectories and venue attributes for next-location prediction [10], whereas Refine-POI

strengthens mobility decisions through reinforcement fine-tuning [14]. However, these methods obtain most of their knowledge from training data, identifiers, or the current prompt. They cannot readily incorporate updated platform relations for an individual request or inject relation-preserving graph tokens into candidate-label generation.

Spatio-temporal mobility modeling. This family learns how visit order, time, and distance constrain the next destination. It prevents a recommender from treating semantically similar venues as equally likely. Flashback handles sparse trajectories through recurrent-state reuse [5], while LSTPM separates long-term preference from short-term intent [6]. GeoSAN introduces geographic distance into self-attention [7]. STAN jointly models spatial and temporal dependencies [8], and DSDRec combines trajectory semantics with region representations [9]. Graph-based alternatives include SNPM, which constructs a dynamic neighborhood graph conditioned on the preceding sequence [23], and MTNet, which organizes check-ins into a mobility tree to learn preferences at multiple temporal granularities [24]. Recent LLM methods express similar constraints through personal and urban context in AgentMove [11], explicit geographic representations in GA-LLM [12], dynamic intent modeling in MFC4POI [13], or geographic reasoning in ROS [15]. Even when these methods combine an LLM with mobility modeling, they typically reduce a trajectory to an ordered token sequence. They do not inject the external relational knowledge implied by that trajectory into generation as structured evidence. Mobility modeling alone also cannot retrieve current external relations, preserve their types, or ensure that they influence competition among valid candidate labels.

External knowledge retrieval and grounding. This family selects facts or subgraphs at inference time, allowing a recommender to use domain knowledge without storing every update in the language model. KAPING retrieves graph facts and turns them into prompt text [16], whereas G-Retriever encodes a task-relevant subgraph to retain connected evidence [17]. Recommendation-specific methods use knowledge graphs in different ways: CoLaKG improves graph comprehension [25], K-RagRec retrieves structured information missing from the LLM [18], and KERAG-R selects relational evidence through GraphRAG [19]. AWARE supplies location- and time-related world knowledge to the next-POI generation [26]. However, existing retrieval objectives do not jointly enforce graph-semantic relevance, user-specific geographic feasibility, and recent intent. Text serialization can also remove typed paths, while a passive graph prefix may not change valid-label probabilities. ASKG addresses both gaps by adapting retrieval to the current trajectory and explicitly connecting relation-preserving graph tokens to candidate-label generation.

Language interfacing, mobility awareness, and knowledge retrieval are each available in isolation. Combining them exposes two recurring gaps. First, continuous spatio-temporal signals must be aligned with the discrete token space of a language model. Second, retrieval must select facts that are both semantically relevant and consistent with the user's mobility range and recent intent. ASKG closes both gaps through adaptive retrieval and relation-preserving graph–language fusion.

3. Problem Definition

Check-in history. Let $\mathcal{U} = \{u_1, \ldots, u_M\}$ and $\mathcal{I} = \{i_1, \ldots, i_N\}$ denote the set of users and POIs, respectively. A timestamped event (u, i, τ) records that user $u \in \mathcal{U}$ visited POI $i \in \mathcal{I}$ at time τ. The interaction data are

$$\mathcal{D} = \{(u, i, \tau) \mid u \in \mathcal{U}, i \in \mathcal{I}\}. \tag{1}$$

For user u, sorting the events by time produces the observed history

$$\mathcal{H}_u = [(i_1, \tau_1), \ldots, (i_T, \tau_T)], \qquad \tau_1 < \cdots < \tau_T. \tag{2}$$

Repeated events may remain within the observed history because revisiting a familiar venue is a valid mobility behavior. The ordered sequence $\mathcal{H}_u$ provides both the adjacent locations used to estimate the user's mobility range and the visit positions used to emphasize recent behavior. The prediction boundary is the final observed event (i_T, τ_T), and the target is the first POI visited after that event. In the reported benchmark construction, target events previously observed for the same user are excluded, so the evaluated next POI is a previously unvisited destination.

Temporal, spatial, and semantic context. Each POI i has a geographic coordinate $\mathbf{g}_i = (\mathrm{lat}_i, \mathrm{lon}_i)$, a category κ_i, and a semantic profile $\mathbf{s}_i$. The profile contains the venue name and the attributes, such as business area, price level, rating, or service tags. We render these fields as short textual descriptions and encode them as $\mathbf{x}_i^{(0)} \in \mathbb{R}^d$. The information sources have different roles in candidate-constrained POI recommendation: coordinates constrain whether a visit is physically plausible, timestamps establish the order of past behavior, categories express functional preference, and semantic descriptions make content accessible to the language model. ASKG keeps these signals distinguishable during retrieval instead of collapsing all of them into text similarity.

POI-specific external knowledge. The external knowledge source is a heterogeneous POI-domain graph

$$\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T}), \qquad \mathcal{T} = \{(e_h, r, e_t) \mid e_h, e_t \in \mathcal{E}, r \in \mathcal{R}\}. \tag{3}$$

Its entities include POIs and the category, business-area, geographic-region, and attribute entities. Its relations describe POI-specific facts such as *has category*, *located in*, *belongs to business area*, and *has attribute*. The Yelp graph additionally contains user–review–business relations derived only from training and validation records; review relations associated with held-out test interactions are excluded during graph construction. The Foursquare graph is automatically derived from the available venue metadata. A mapping set $\mathcal{M} = \{(i, e) \mid i \in \mathcal{I}, e \in \mathcal{E}\}$ aligns POIs with their graph entities. The graph serves as external venue evidence rather than a prediction target or a generic collection of world facts. For example, two POIs connected to the same category or business area remain associated even when their descriptions use different words. ASKG retrieves local subgraphs from the user's history and conditions their use on the current candidate prompt.

Candidate-constrained generative objective. For a prediction instance, the input consists of the user's history $\mathcal{H}_u$, the profiles of a supplied candidate set $C_u = \{c_1, \ldots, c_n\} \subseteq \mathcal{I}$, and the external POI knowledge $\mathcal{G}$. Exactly one candidate $i^\star \in C_u$ is the observed next POI. In this benchmark, neither $i^\star$ nor any negative candidate appears in the user's earlier visits, and every compared method uses the same history and candidate set. Access to the same external knowledge source does not imply identical retained facts or input representations; these differences are discussed in Section 5.1. The benchmark constructs C_u with the target included and does not evaluate an upstream candidate generator. Each candidate c_j receives a unique temporary label a_j. The labels and descriptions, together with the task instruction, form the textual prompt $\mathbf{q}_u$. ASKG encodes the history separately as an ordered POI segment so that it is not duplicated in its prompt.

ASKG retrieves POI evidence from $\mathcal{G}$ using $\mathcal{H}_u$, represents the history, retrieved evidence, and candidates as three graph-token segments, and grounds the prompt in these segments. Let ℓ_j denote the next-token logit of candidate label a_j. The model predicts the candidate index $\hat{j}$ with the highest logit, from which it reads off the generated candidate label $\hat{a}$ (the textual token, e.g., "A") and the corresponding recommended POI $\hat{\imath}$ (the venue that $\hat{a}$ denotes in the current candidate set):

$$\hat{j} = \arg\max_{j \in \{1,\ldots,n\}} \ell_j, \qquad \hat{a} = a_{\hat{j}}, \qquad \hat{\imath} = c_{\hat{j}}. \tag{4}$$

The labels are drawn from a tokenizer-compatible option alphabet large enough for the current candidate set, and all option logits are read at the first answer position. Their softmax probabilities define the candidate ranking. The model does not generate the candidate set: the task is *candidate-constrained*, and its output is the label associated with the most plausible next visit. Table 1 summarizes the notation introduced in this section and used throughout the remainder of this paper.

Table 1. Main notation and its meaning.

Symbol	Meaning
$\mathcal{U}, \mathcal{I}$	User set, POI set ($\|\mathcal{U}\|=M$, $\|\mathcal{I}\|=N$)
$\mathcal{D}$	Timestamped user–POI check-in data
$\mathcal{H}_u, C_u$	Timestamped history and n-option candidate set of user u
$\mathbf{g}_i, \kappa_i, \mathbf{s}_i$	Coordinate, category, and semantic profile of POI i
$\mathbf{x}_i^{(0)}$	Initial sentence embedding of POI i's description
$\mathcal{G}, \mathcal{E}, \mathcal{R}, \mathcal{T}$	POI-domain graph, entities, relations, and triples
$\mathcal{M}, \mathbf{e}_r$	POI–entity mapping and embedding of relation type r
$L, \mathbf{X}^{(L)}$	Propagation depth and multi-hop relation-aware POI retrieval index
$\bar{d}_{\text{seq}}, \alpha$	Mean adjacent travel distance and adaptive geographic weight
$D_{\max}, \alpha_{\min}, \alpha_{\max}$	Mobility-distance cap and geographic-weight bounds
b, n_g, n_{geo}	Per-POI recall budget and graph/geographic quotas
$\mathcal{R}_{\text{graph}}, \mathcal{R}_{\text{geo}}, \mathcal{P}(i)$	Graph recall, geographic recall, and merged recall set for POI i
$\mathcal{U}_u, \lambda, \mathbf{q}_{\text{dyn}}$	Recalled evidence pool, sequence-position decay, and dynamic-preference query
$w_k, s_{\text{graph}}, s_{\text{geo}}, \phi$	Sequence-position weight and graph, geographic, and fused evidence scores
r_{geo}, m	Geographic influence radius and retained evidence count
$\mathcal{S}_u$	Top-m evidence seed nodes $\{v_1, \ldots, v_m\}$ retained after re-ranking
$\mathbb{G}_u$	Evidence subgraph set $\{\mathcal{G}_1, \ldots, \mathcal{G}_m\}$ expanded from $\mathcal{S}_u$
$N_s, H_a, \mathbf{H}_s$	Node count, attention-head count, and encoded features of subgraph s
$\mathbf{Q}, \mathbf{Z}_s, K$	Shared SBF queries, compressed tokens of subgraph s, and tokens per subgraph
$\mathbf{G}_h, \mathbf{G}_e, \mathbf{G}_c$	History, retrieved-evidence, and candidate graph-token segments
p_t	Query-aware gating score
$\tilde{\mathbf{G}} = [\tilde{\mathbf{G}}_h; \tilde{\mathbf{G}}_e; \tilde{\mathbf{G}}_c]$	Complete gated three-segment graph sequence
$L_q, \mathbf{T}, \mathbf{T}^{\text{out}}$	Prompt length and prompt-token embeddings before and after graph–language interaction
$\mathbf{q}_u, \mathbf{E}_{\text{in}}, \theta$	Textual prompt, final input embeddings, and LLM parameters
$\mathbf{e}_{\text{sep}}, \gamma, \rho_w, \eta$	Segment separator, residual gate, graph warm-up factor, and LoRA learning-rate ratio
$a_j, j^*, \hat{a}, \ell_j$	Candidate option label, target-label index, generated label, and option logit
$i^*, \hat{i}$	Observed and predicted next POI
$\tau_c, \alpha_{\text{loss}}$	Candidate temperature and generation-loss weight
$\mathcal{L}_{\text{gen}}, \mathcal{L}_{\text{disc}}, \mathcal{L}$	Generation, candidate-discrimination, and total losses

4. Methodology

Figure 1 illustrates how ASKG turns external POI evidence into a recommendation. It first finds evidence that is both related to the user's interests and geographically plausible, then preserves the relations within that evidence, and finally incorporates the evidence into the candidate prompt for candidate-label comparison. The first module, Adaptive Spatio-Temporal Evidence Retrieval, builds a relation-aware POI index, allocates graph-semantic and geographic recall according to the user's mobility range, and re-ranks the evidence pool toward recent intent. Relation-Preserving Graph–Language Fusion then encodes typed local subgraphs and compresses them into a fixed token budget while keeping the history, evidence, and candidates separate. Candidate-Oriented POI Generation and Alignment connects these representations to the prompt and maps the selected label back to a POI. Progressive training allows the graph pathway to add useful evidence without abruptly changing the pretrained language representation.

4.1. Adaptive Spatio-Temporal Evidence Retrieval

Text-similarity retrieval can favor a semantically attractive POI outside the user's travel range. ASKG addresses this limitation with a three-stage retrieval pipeline. First, it builds a *multi-hop relation-aware POI index* in which closeness reflects both descriptive similarity and graph relations. Second, it performs *adaptive graph-semantic and geographic recall*. This

dual-path search allocates two recall paths according to the user's mobility pattern rather than a fixed ratio. Third, ASKG applies *recency-aware evidence re-ranking* so that later visits in the trajectory have more influence than earlier ones. Before retrieval, ASKG selects the least-frequent half of the POIs in the observed history using corpus-level counts from the preprocessing pipeline. Frequent venues already have comparatively strong behavioral support, whereas long-tail venues benefit more from external relations. This selection controls retrieval cost and determines the history anchors. The complete ordered history remains available in the history-token segment and the recency-aware global query.

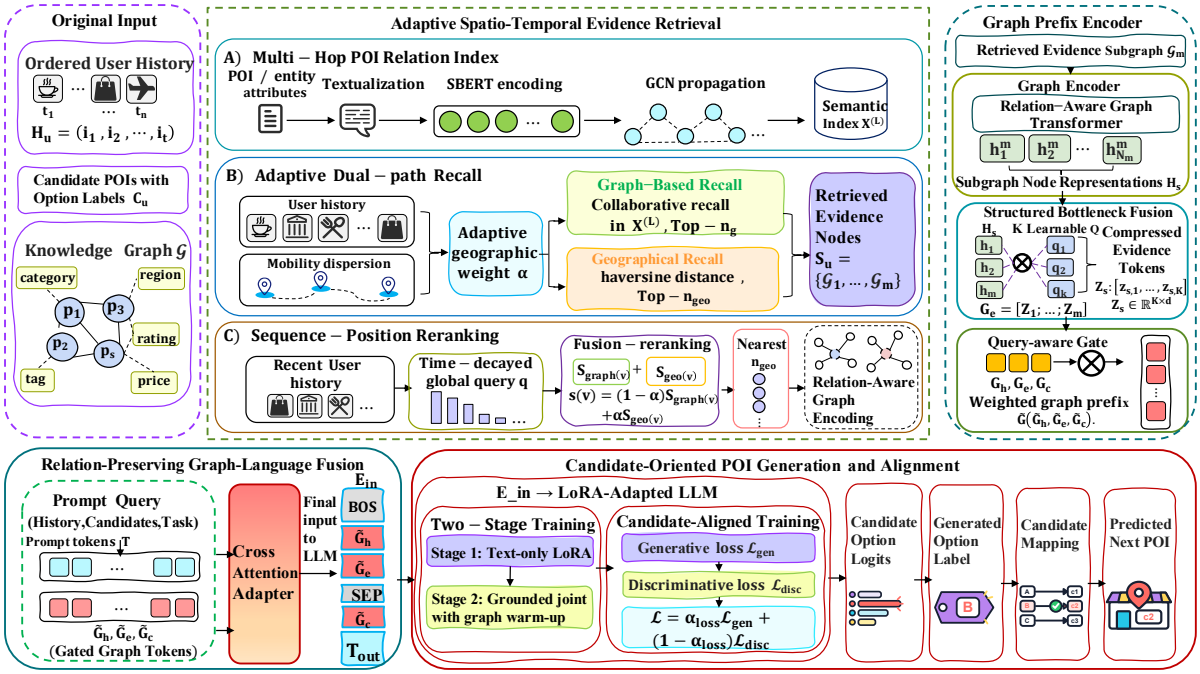


Figure 1. ASKG for candidate-constrained POI recommendation. Retrieval balances graph-semantic and geographic evidence and re-ranks it according to the user's recent trajectory before expanding the retained seeds $\mathcal{S}_u = \{v_1, \ldots, v_m\}$ into typed subgraphs $\mathbb{G}_u$. Fusion represents the history, evidence, and candidates as separate graph-token segments and lets the prompt query them through cross-attention. A LoRA-adapted LLM then produces option logits, generates a label, and maps it to the predicted next POI. The lower path shows the two-stage schedule and candidate-aligned objective used to train this evidence-to-generation process.

Multi-hop relation-aware POI index. Text embeddings compare venues independently and may miss functional alternatives whose descriptions differ. Graph connectivity alone, however, does not express the semantic content of a category or attribute. We therefore construct an index in which closeness reflects both descriptive similarity and POI-domain relations. Each relation type r has an edge embedding $\mathbf{e}_r$. Direct POI–POI edges are not required because relation paths connect venues through shared categories, business areas, geographic containment, or other attributes. Let $\mathcal{N}(i)$ contain pairs (j, r_{ji}) that identify a neighboring entity and its edge type. We construct the index with L layers of relation-aware normalized propagation:

$$\mathbf{x}_i^{(l+1)} = \mathrm{ReLU}\left(\mathbf{W}_0^{(l)} \mathbf{x}_i^{(l)} + \frac{1}{|\mathcal{N}(i)|} \sum_{(j,r_{ji}) \in \mathcal{N}(i)} \left(\mathbf{W}_n^{(l)} \mathbf{x}_j^{(l)} + \mathbf{W}_r^{(l)} \mathbf{e}_{r_{ji}} \right) \right), \tag{5}$$

where $\mathbf{e}_{r_{ji}}$ is the embedding of relation type r_{ji}, and $\mathbf{W}_0^{(l)}$, $\mathbf{W}_n^{(l)}$, and $\mathbf{W}_r^{(l)}$ are learnable transformations. Thus, messages arriving through category, business-area, attribute, and geographic relations remain distinguishable during propagation. After L layers, $\mathbf{x}_i^{(L)}$ summarizes the entity's L-hop typed relational neighborhood rather than a fixed path template. Shorter

paths connect venues that share a category or business area. Deeper propagation reaches broader category and region structures but may smooth distinct POIs toward a common regional representation. The resulting index $\mathbf{X}^{(L)} \in \mathbb{R}^{|\mathcal{E}|\times d}$ is therefore relation-aware rather than merely text-similar. The propagation depth used in the reported experiments is given in Section 5.1. Geographic distance remains a separate signal so that multi-hop structural similarity does not obscure physical feasibility.

Adaptive graph-semantic and geographic recall. The multi-hop index finds POIs that are functionally or structurally related, but it cannot determine whether they are reachable for a particular user. A compact history calls for stronger geographic filtering than a trajectory spread across distant regions. A fixed allocation can either miss nearby alternatives or over-restrict a mobile user. ASKG instead uses parallel graph-semantic and geographic recall paths controlled by a *user-level adaptive geographic weight* α. Let $\bar{d}_{\text{seq}}$ be the mean haversine distance between adjacent POIs in the observed trajectory. The adaptive weight α is formally defined as

$$\alpha = \text{clip}\left(\alpha_{\max} - (\alpha_{\max} - \alpha_{\min})\frac{\min(\bar{d}_{\text{seq}}, D_{\max})}{D_{\max}},\ \alpha_{\min},\ \alpha_{\max}\right). \tag{6}$$

When the user's activity range is concentrated, the geographic path receives a larger recall quota. As the mean adjacent distance approaches the mobility cap $D_{\max}$, the graph-semantic path receives more weight. The same value is used in recall allocation and evidence re-ranking. Histories with fewer than two geocoded POIs receive a fixed default weight because $\bar{d}_{\text{seq}}$ is undefined under this scenario.

The weight α divides the recall budget between two independent paths for each historical POI i. The *graph-semantic path* retrieves the n_g nearest nodes in the propagated embedding space $\mathbf{X}^{(L)}$ by cosine similarity:

$$\mathcal{R}_{\text{graph}}(i) = \underset{v\in\mathcal{I}\setminus\{i\}}{\text{Top-}n_g}\ \cos\big(\mathbf{X}^{(L)}[i],\ \mathbf{X}^{(L)}[v]\big), \tag{7}$$

Because $\mathbf{X}^{(L)}$ contains attribute-bridged relations, this path can discover both descriptively similar POIs and venues connected through shared category or business-area entities. The *geographic-proximity path* remains independent of the embedding space and recalls the nearest n_{geo} POIs by haversine distance:

$$\mathcal{R}_{\text{geo}}(i) = \underset{v\in\mathcal{I}\setminus\{i\}}{\text{Bottom-}n_{\text{geo}}}\ d_{\text{haversine}}(i, v), \tag{8}$$

The fixed per-POI recall budget b is divided into two quotas:

$$n_{\text{geo}} = \text{clip}(\text{round}(b\alpha), 1, b-1), \qquad n_g = b - n_{\text{geo}}. \tag{9}$$

Keeping the geographic path independent prevents relation similarity from acting as a proxy for distance. A nearby venue from a different category can still be a practical alternative even when a structurally similar venue lies far outside the user's activity range. The two result sets are merged and deduplicated to obtain $\mathcal{P}(i) = \mathcal{R}_{\text{graph}}(i) \cup \mathcal{R}_{\text{geo}}(i)$, which is then passed to sequence-position re-ranking.

Recency-aware evidence re-ranking. Dual-path recall produces local evidence around multiple historical POIs. To represent the changing visit intent, we form a sequence-position-weighted global query. For the history in Equation (2), the POI graph embeddings are aggregated as

$$w_k = \frac{\lambda^{T-k}}{\sum_{j=1}^{T}\lambda^{T-j}}, \quad \mathbf{q}_{\text{dyn}} = \sum_{k=1}^{T} w_k \mathbf{X}^{(L)}[i_k], \tag{10}$$

where $\lambda \in (0,1)$ is the sequence-position decay factor. The exponent $T-k$ measures how many visits separate i_k from the end of the sequence rather than elapsed clock time. Because the trajectory is ordered chronologically, the last position is the most recent visit, so the exponential decay assigns it the largest weight in $\mathbf{q}_{\text{dyn}}$. We refer to this scheme as recency-aware.

Using $\mathbf{q}_{\text{dyn}}$ as the global query, for each node v in the recalled evidence pool $\mathcal{V}_u = \bigcup_{k=1}^{T} \mathcal{P}(i_k)$, we compute a graph score and a geographic score:

$$s_{\text{graph}}(v) = \cos(\mathbf{q}_{\text{dyn}}, \mathbf{X}^{(L)}[v]), \tag{11}$$

$$s_{\text{geo}}(v) = \max\big(0,\ 1 - d_{\text{haversine}}(h_u(v), v)/r_{\text{geo}}\big), \tag{12}$$

where $h_u(v)$ records the historical POI from which v was retrieved, and r_{geo} is the geographic influence radius. The geographic score decays linearly to zero at r_{geo}. To ensure scale consistency between two scores, each score is max-normalized within $\mathcal{V}_u$ as $\bar{s}_x(v) = s_x(v)/\max(\max_{w\in\mathcal{V}_u} s_x(w), \epsilon)$ and fused with the same user-specific weight:

$$\phi(v) = (1-\alpha)\,\bar{s}_{\text{graph}}(v) + \alpha\,\bar{s}_{\text{geo}}(v). \tag{13}$$

We retain the m nodes with the highest fused scores as $\mathcal{S}_u = \{v_1, \ldots, v_m\}$. Expanding the first-order neighborhood of each seed, including its attribute entities and typed edges, produces $\mathbb{G}_u = \{\mathcal{G}_1, \ldots, \mathcal{G}_m\}$ for the fusion stage.

4.2. Relation-Preserving Graph–Language Fusion

Candidate-constrained POI recommendation combines the retrieved evidence with an ordered history and the candidate POIs. ASKG represents them as three graph-token segments: one token per observed POI, a fixed number per evidence subgraph, and one token per candidate. A Graph Transformer encodes the singleton POI graphs and evidence subgraphs with typed edges, after which the Subgraph Bottleneck Former compresses the variable-size evidence. Query-aware gating weights the three segments against the prompt, and a cross-attention adapter lets its representations consult the gated graph tokens. The resulting interface preserves visit order and candidate identity while reserving additional capacity for relational evidence.

Relation-aware encoding and bottleneck compression. During retrieval, a node embedding captures an entity's identity but not its role in the selected subgraph. A POI linked to both a category and a business area carries a configuration that independent node vectors do not express. We encode it with a Graph Transformer [27] using multi-head edge-aware attention:

$$\mathbf{h}_i^{(l+1)} = \Big\Vert_{r=1}^{H_a} \left(\mathbf{W}_1^{(r)} \mathbf{h}_i^{(l)} + \sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(r)} \big(\mathbf{W}_2^{(r)} \mathbf{h}_j^{(l)} + \mathbf{W}_e^{(r)} \mathbf{e}_{r_{ji}}\big) \right), \tag{14}$$

where H_a is the number of attention heads, $\mathbf{e}_{r_{ji}}$ is the embedding of edge type r_{ji} connecting j to i, and $\alpha_{ij}^{(r)}$ is the corresponding edge-aware attention weight. The update can therefore distinguish a category neighbor from a geographic neighbor instead of treating the subgraph as an unordered set. Parameters are shared across all singleton graphs and evidence subgraphs, yielding $\mathbf{H}_s \in \mathbb{R}^{N_s \times d}$ for evidence subgraph s.

Fixed-budget subgraph compression. Even after relation-aware encoding, the m selected subgraphs vary in size and can contain hundreds of node tokens, so the LLM cannot effectively leverage this structural information until it is compressed to a fixed length.

Inspired by Q-Former [28], the Subgraph Bottleneck Former (SBF) uses K learnable queries to convert each subgraph into a fixed number of tokens:

$$\mathbf{Z}_s = \mathrm{CrossAttn}(\mathbf{Q},\ \mathbf{H}_s,\ \mathbf{H}_s) \in \mathbb{R}^{K\times d}, \tag{15}$$

where $\mathbf{Q} \in \mathbb{R}^{K\times d}$ is shared across subgraphs. Cross-attention learns which encoded nodes are needed for recommendation without using indiscriminate mean pooling. Concatenating the compressed evidence gives $\mathbf{G}_e = [\mathbf{Z}_1; \ldots; \mathbf{Z}_m] \in \mathbb{R}^{mK\times d}$.

The history and candidate POIs are represented as singleton graphs and do not require bottleneck compression. After Graph Transformer encoding and projection to the LLM dimension, their tokens are

$$\mathbf{G}_h = [\mathbf{g}_1^h; \ldots; \mathbf{g}_T^h] \in \mathbb{R}^{T\times d}, \qquad \mathbf{G}_c = [\mathbf{g}_1^c; \ldots; \mathbf{g}_n^c] \in \mathbb{R}^{n\times d}. \tag{16}$$

The ordered graph-token sequence is $\mathbf{G} = [\mathbf{G}_h; \mathbf{G}_e; \mathbf{G}_c]$. Thus, each POI in the history and candidate segments is represented by a single token, while the external evidence uses mK tokens regardless of the original subgraph sizes.

Query-conditioned graph–language interaction. The three segments contain complementary information, but not every token is equally useful for the current candidate set. Let $\mathbf{T} = \mathrm{Embed}_\theta(\mathbf{q}_u) \in \mathbb{R}^{L_q\times d}$ be the embeddings of the task instruction, candidate labels, and their corresponding descriptions with $\bar{\mathbf{t}}$ denoting their mean. For graph token $\mathbf{g}_t$, query-aware gating combines the token, the prompt summary, and their cosine similarity:

$$p_t = \sigma\big(\mathrm{MLP}_{\mathrm{gate}}([\mathbf{g}_t \,\|\, \bar{\mathbf{t}} \,\|\, c_t])\big), \quad c_t = \cos(\mathbf{g}_t, \bar{\mathbf{t}}). \tag{17}$$

The score $p_t \in (0,1)$ scales $\mathbf{g}_t$ to $\tilde{\mathbf{g}}_t = p_t \mathbf{g}_t$. Retaining the original segment boundaries yields $\tilde{\mathbf{G}} = [\tilde{\mathbf{G}}_h; \tilde{\mathbf{G}}_e; \tilde{\mathbf{G}}_c]$. The gate can therefore suppress an unhelpful history, evidence, or candidate token without discarding the three-segment organization.

Explicit cross-attention. Prefix concatenation makes the three graph segments visible to the LLM, but relying only on pretrained self-attention does not guarantee that the textual candidate representations will use them. We add an explicit interaction path in which the prompt tokens query the complete gated graph sequence $\tilde{\mathbf{G}}$:

$$\mathbf{T}^{\mathrm{enh}} = \mathrm{CrossAttnAdapter}(\mathbf{T},\ \tilde{\mathbf{G}}). \tag{18}$$

To avoid disrupting the LLM's existing language-modeling ability early in training, the adapter's output is injected into the original text representation in a residual manner via a learnable scalar gate γ. Specifically, the graph-information increment $\Delta = \mathbf{T}^{\mathrm{enh}} - \mathrm{LN}(\mathbf{T})$ is computed in the LayerNorm-normalized space and then added back to the original text representation:

$$\mathbf{T}^{\mathrm{out}} = \mathbf{T} + \gamma \cdot \Delta \cdot \rho_w, \quad \gamma = \sigma(w_{\mathrm{gate}}), \tag{19}$$

where $\rho_w \in [0,1]$ increases linearly during the warm-up steps of joint fine-tuning and remains 1 afterward. Initializing w_{gate} to a small negative value keeps γ close to zero at the start of joint training. The graph contribution therefore grows gradually instead of abruptly changing the LLM input distribution, while the residual form preserves the original text representation.

4.3. Candidate-Oriented POI Generation and Alignment

The overall training objective is a weighted combination of two cross-entropy terms: standard language-modeling entropy over the full vocabulary and a candidate-restricted entropy that forces probability mass to concentrate on the n valid options.

The final module converts the three grounded graph segments into a candidate-label prediction and aligns training with that space because ordinary language-model training distributes probability over the entire vocabulary, whereas the recommendation compares only the n valid labels.

Three-segment generative input. The gated history, evidence, and candidate segments retain the order established in Equation (16). A learned separator $\mathbf{e}_{\text{sep}} \in \mathbb{R}^{1\times d}$ marks the boundary before the candidate segment. It is shown as SEP in Figure 1. The LLM input is

$$\mathbf{E}_{\text{in}} = [\mathbf{e}_{\text{BOS}} \,\|\, \tilde{\mathbf{G}}_h \,\|\, \tilde{\mathbf{G}}_e \,\|\, \mathbf{e}_{\text{sep}} \,\|\, \tilde{\mathbf{G}}_c \,\|\, \mathbf{T}^{\text{out}}] \in \mathbb{R}^{(T+mK+n+L_q+2)\times d}. \tag{20}$$

The graph sequence is used in two complementary ways. It remains in the prefix as explicit tokens, providing the keys and values for the cross-attention that derives $\mathbf{T}^{\text{out}}$. Each candidate is assigned a tokenizer-compatible candidate label. At the first answer position, the implementation extracts the tokenizer ID used for each valid label and ranks the corresponding logits $\ell_1, \ldots, \ell_n$. It does not freely decode a venue name. The prediction is

$$\hat{j} = \arg\max_{j\in\{1,\ldots,n\}} \ell_j, \qquad \hat{a} = a_{\hat{j}}, \qquad \hat{i} = c_{\hat{j}}. \tag{21}$$

Two-stage training. *Stage 1: Text-only LoRA warm-up.* The graph pathway is skipped, its parameters are frozen, and only the LLM's LoRA parameters are updated [29]. The input contains the task instruction, candidate labels, and descriptions but no graph prefix, allowing the backbone to learn the output format before receiving graph representations. *Stage 2: Grounded joint fine-tuning*. The graph encoder, bottleneck, projector, gates, and cross-attention adapter are unfrozen and optimized with LoRA. During the initial steps, a coefficient ρ_w scales the projected graph tokens and cross-attention increment from 0 to 1. The LoRA learning rate is η times the graph-module rate, with $\eta = 0.3$ for Qwen and 0.5 for the other backbones.

Candidate-aligned generation loss. Let j^* be the index of the observed next POI and let $\mathbf{y} = (a_{j^*}, \text{EOS})$ be the short target sequence. The full-vocabulary generation loss is

$$\mathcal{L}_{\text{gen}} = -\sum_{t=1}^{|\mathbf{y}|} \log P_\theta(y_t \mid \mathbf{E}_{\text{in}}, y_{<t}), \tag{22}$$

which preserves the language model's answer-generation behavior and encourages a format-valid output.

To enhance discrimination among the valid candidates beyond what full-vocabulary generation alone provides, we additionally normalize only the n option logits at the first answer position since tokens outside the option set cannot denote a candidate POI:

$$\mathcal{L}_{\text{disc}} = -\log \frac{\exp(\ell_{j^*}/\tau_c)}{\sum_{j=1}^{n} \exp(\ell_j/\tau_c)}, \tag{23}$$

where τ_c is the candidate-softmax temperature and equals 1 in the reported configuration. The training objective used in the reported experiments is

$$\mathcal{L} = \alpha_{\text{loss}}\mathcal{L}_{\text{gen}} + (1-\alpha_{\text{loss}})\mathcal{L}_{\text{disc}}, \tag{24}$$

where α_{loss} balances output-format learning and candidate discrimination. No contrastive or auxiliary graph-ranking term is included in the reported experiments.

Algorithm 1 summarizes the complete grounded training and inference flow after the module-level formulation above. Stage 1 is the text-only warm-up described in Section 4.3 and therefore skips the graph pathway.

Algorithm 1 ASKG: Grounded Training and Inference for Candidate-Label Generation

Require: History $\mathcal{H}_u$, candidates $\mathcal{C}_u$, prompt $\mathbf{q}_u$, POI graph $\mathcal{G}$, index $\mathbf{X}^{(L)}$
Ensure: Training loss $\mathcal{L}$ or generated label $\hat{a}$ and recommended POI $\hat{i}$
Select low-frequency history anchors; compute α (Equation (6)) and allocate recall budget b
for each selected historical POI i_k **do**
  Retrieve graph-similar and geographically close POIs; merge them into $\mathcal{U}_u$
end for
Form $\mathbf{q}_{\text{dyn}}$, re-rank $\mathcal{U}_u$, retain $\mathcal{S}_u$, and expand typed subgraphs $\mathbb{G}_u$
Encode history and candidate singleton graphs as $\mathbf{G}_h$ and $\mathbf{G}_c$
for each evidence subgraph $\mathcal{G}_s \in \mathbb{G}_u$ **do**
  $\mathbf{H}_s \leftarrow \text{GraphTransformer}(\mathcal{G}_s)$; $\mathbf{Z}_s \leftarrow \text{SBF}(\mathbf{H}_s)$ (Equation (15))
end for
Form $\mathbf{G}_e$; gate the three graph segments (Equation (17)) and obtain $\mathbf{T}^{\text{out}}$ (Equation (19))
Assemble $\mathbf{E}_{\text{in}}$ (Equation (20)) and obtain valid option logits $\{\ell_1, \dots, \ell_n\}$
if training **then**
  Return $\mathcal{L} = \alpha_{\text{loss}}\mathcal{L}_{\text{gen}} + (1 - \alpha_{\text{loss}})\mathcal{L}_{\text{disc}}$ (Equation (24))
end if
$\hat{j} \leftarrow \arg\max_j \ell_j$; $\hat{a} \leftarrow a_{\hat{j}}$; $\hat{i} \leftarrow c_{\hat{j}}$ (Equation (21))
return $(\hat{a}, \hat{i})$

5. Experiments

We conduct experiments to evaluate ASKG and compare it with the baselines. The analysis is organized around four research questions, with the full metric set (ACC, Recall@3/5, NDCG@3/5/10, and MRR) reported for all experimentsunless stated otherwise:

- **RQ1: Overall Effectiveness.** Does ASKG improve candidate-label generation over text-injection and retrieval-augmented baselines under matched candidate sets and training budgets?
- **RQ2: Component Contribution.** Which parts of ASKG produce this improvement? We replace its retrieval, relation-preserving fusion, cross-attention, and candidate-aligned discrimination one at a time and trace the resulting change in ranking quality.
- **RQ3: Backbone and Relational-Density Robustness.** Does the benefit of grounding hold across language models of different scales and pretraining sources and across graphs with different relational densities?
- **RQ4: Training Sensitivity.** How does the Stage 2 LoRA learning-rate ratio affect ASKG's recommendation performance?

5.1. Experimental Setup

Datasets and evaluation protocol. We conduct our experiments on the Yelp Open Dataset [30] and the Foursquare NYC check-in dataset [31]. Yelp ratings of at least four stars are treated as positive interactions, while every Foursquare check-in is positive. Interactions are ordered chronologically for each user. The latest interaction is held out for testing, the preceding interaction is used for validation, and the remainder forms the training history. Both knowledge graphs contain POI categories and geographic relations. Yelp additionally provides business-area, attribute, and category-hierarchy relations. After preprocessing, we evaluate all 1083 eligible Foursquare users and a fixed subset of 1000 Yelp users. The fixed Yelp subset controls the cost of LLM generation while preserving paired comparisons across methods. It is sampled uniformly without replacement using seed 9. The same numeric seed also initializes parameters and controls data loading; candidate construction and model training are nevertheless treated as conceptually distinct random sources since one

determines the sampled user subset and option pools while the other determines parameter initialization and data order. Table 2 reports statistics of the two processed datasets.

Table 2. Statistics of the processed datasets.

Statistic	Yelp	Foursquare NYC
Users	19,913	1083
POIs	32,502	38,333
Interactions	1,120,850	227,428
Test instances S	1000	1083
KG entities	35,005	40,053
KG relations	15	4
KG triples	640,156	99,196

We follow a candidate-constrained generation protocol. Each instance contains the observed next POI and a fixed mixture of hard negatives, after which the candidate order is randomly permuted. Yelp uses nine negatives: four share the first normalized token of the venue name with the target, three come from the target's geographic grid neighborhood, and two come from the same or an adjacent interaction-frequency bucket. Foursquare uses 19 negatives: eight share at least one category with the target, six come from its geographic grid neighborhood, and five come from the same or an adjacent frequency bucket.

For both datasets, valid POIs are sorted by global interaction count and divided into ten equal-cardinality frequency buckets. Geographic pools use a 0.01-degree grid and include cells within five grid steps of the target in each coordinate. Candidates are sampled uniformly without replacement from each eligible pool. Any shortfall is filled uniformly from all valid POIs. The preprocessing excludes targets previously visited by the user, and every negative differs from both the target and the user's earlier POIs. Training, validation, and testing use the same construction rule. All methods receive the same candidate sets within a dataset. The resulting alternatives are related by name or category, geographic locality, or interaction frequency. No global maximum-distance filter is imposed on candidates drawn from the other pools.

The language model scores the option labels at the first answer position. Yelp uses A–J for 10 options, and Foursquare uses A–T for 20 options across all 1083 test instances. We normalize the corresponding logits over the valid labels to obtain a candidate ranking. ACC records whether the highest-scoring label denotes the observed next POI; Recall@3, Recall@5, NDCG@3, NDCG@5, NDCG@10, and MRR are computed from the same ranking. The resulting protocol is candidate-constrained: the model generates a label associated with one of the supplied POIs.

The saved tokenizer check covers Qwen2.5-7B, LLaMA-3.1-8B, GLM-4-9B, and Gemma-2-9B. Each bare capital letter A–T maps to one distinct token within its tokenizer; Appendix A gives every ID and the checked model identifiers. The recorded checks compare bare labels and labels following the assistant-turn marker. Leading-space variants have different IDs, so the bare-letter target and the actual answer boundary must remain consistent. A shared one-token interface avoids label-length differences but does not establish equal prior probabilities across letters or eliminate all label-position bias. Random candidate ordering removes a fixed POI–letter mapping rather than proving tokenizer neutrality. Candidate order was randomized during candidate-set construction and then held fixed for the reported paired comparisons; we did not re-evaluate the same fixed candidate set under multiple candidate-to-label permutations, so robustness to candidate-label permutation and invariance to letter-specific priors remain untested.

The first-position readout is defined only for single-token labels. A tokenizer that splits a proposed label would require remapping to distinct validated one-token labels. If a multi-token output were deliberately retained, its full conditional sequence probability, rather than only its first-token logit, would be needed, with an explicit policy for length effects. This is an extension, not an implemented capability established by the present results.

The supplied candidate sets always contain the observed target. They measure ranking conditional on target inclusion and do not measure the recall of an upstream POI-candidate generator. System-level candidate generation and the graph-evidence retrieval budget in Section 4.1 are separate operations.

Let r_s be the rank of the observed next POI for test instance s, and let S be the number of test instances. Because each instance contains one relevant candidate, the metrics are defined as

$$\begin{aligned} \text{ACC} &= \frac{1}{S}\sum_{s=1}^{S}\mathbb{I}[r_s = 1], \\ \text{Recall@}K &= \frac{1}{S}\sum_{s=1}^{S}\mathbb{I}[r_s \le K], \\ \text{NDCG@}K &= \frac{1}{S}\sum_{s=1}^{S}\frac{\mathbb{I}[r_s \le K]}{\log_2(r_s+1)}, \\ \text{MRR} &= \frac{1}{S}\sum_{s=1}^{S}\frac{1}{r_s}, \end{aligned} \tag{25}$$

where S is the number of held-out test instances: 1000 for Yelp and 1083 for Foursquare.

Baselines and implementation. KGText supplies retrieved evidence as natural-language prompt text; KAPING [16] is adapted to the candidate-label task using a triple-style evidence prompt; and K-RagRec [18] supplies a recommendation-oriented retrieval baseline. The released KGText and KAPING templates contain a task instruction, an evidence block, the visiting history, the candidate list, and a single-letter output instruction. The current serializer expresses a shared-attribute POI path as a triple in KAPING and as a sentence in KGText. For example, the same path can be rendered as `(POI-1, coVisitedWith[category], POI-2)` or as a sentence linking POI-1 and POI-2 through that shared category. The relation name in this template is a serialization convention and should not be taken as additional observed co-visitation evidence.

These controls align the task and candidate-label interface, not every aspect of information access. Using the same graph source does not imply identical retrieved facts, attribute detail, or retained information after truncation. In particular, the released text-baseline path supplies retrieved history evidence and does not separately inject candidate subgraphs; ASKG includes a candidate segment. Consequently, the comparison concerns the implemented evidence pipelines rather than a fully isolated test of graph encoding alone. LoRA adaptation also does not establish that a text baseline has received optimal prompt formatting.

Each dataset uses three backbones: Qwen2.5-7B and LLaMA-3.1-8B on both datasets, GLM-4-9B on Yelp, and Gemma-2-9B on Foursquare. The configuration reported in this draft uses LoRA, a common five-epoch ceiling, and two text-only plus three grounded epochs for ASKG. The reported graph settings are a four-layer encoder, eight bottleneck tokens per retrieved subgraph, and two cross-attention layers.

Table 3 reports the core configuration needed to interpret and reproduce the comparison. We use AdamW and divide the common five-epoch budget into two text-only epochs and three grounded epochs. The LoRA learning rate in Stage 1 is 1×10^{-5}. During Stage 2, it is set to 0.3 times the graph-module rate for Qwen and 0.5 times that rate for the other backbones. The graph prefix is introduced gradually over the first 500 Stage 2 steps. This warm-up prevents newly initialized graph modules from abruptly altering the pretrained

language representation. Remaining backbone-specific optimizer details follow the shared implementation and do not change the retrieval budget, graph-token capacity, or candidate protocol reported in the table.

Table 3. Core experimental configuration.

Setting	Reported Value
Candidate-set size (Yelp/Foursquare)	10/20
Prompt length/effective batch size	512/32
Optimizer/graph learning rate	AdamW/1×10^{-5}
Training budget/stage split	5 epochs/2 + 3 epochs
LoRA rank (Qwen, LLaMA, GLM/Gemma)	16/8
Graph heads/bottleneck tokens/adapter layers	4/8/2
Index propagation depth L	2
Retrieval anchors	Least-frequent 50% of history
Recall quota/retained evidence subgraphs	12/10
Mobility cap/geographic radius	15 km/5 km
Geographic-weight range/default weight	$[0.2, 0.5]$/0.3
Sequence-position decay λ	0.8
Generation-loss weight/candidate temperature	0.05/1.0

5.2. Main Results

This section answers RQ1 by comparing ASKG against the three baselines on both datasets. Table 4 reports the full metric set for all six dataset–backbone settings.

Yelp. ASKG improves both the generated first choice and the rank of the observed next POI with all three backbones. Against K-RagRec, the strongest baseline on ACC and MRR, the MRR gains are 7.5%, 9.1%, and 4.1% for Qwen, LLaMA, and GLM, while ACC rises by 11.3%, 10.6%, and 12.4%. These two metrics describe complementary aspects of the same generative decision: ACC records whether the correct candidate label is generated first, whereas MRR also rewards moving that label toward the top when it is not the first choice. The gains extend to deeper ranks for Qwen and LLaMA but not uniformly for GLM. In particular, KGText retains a small advantage on GLM R@5 (0.7380 versus 0.7300), suggesting that ASKG's clearest contribution in this setting is to improve the ordering near the top of the candidate list.

Foursquare. On the complete set of 1083 held-out instances under the 20-option protocol, ASKG achieves the highest MRR with all three backbones, with gains of 7.0% for Qwen, 6.0% for LLaMA, and 1.7% for Gemma over the strongest MRR baseline. The cutoff metrics, however, are less favorable than on Yelp: with LLaMA, ASKG's ACC is 4.6% *below* K-RagRec, and KGText or KAPING is stronger than ASKG at several Recall and NDCG cutoffs across all three backbones. The Foursquare knowledge graph contains four relation types, compared with fifteen for Yelp. Consequently, its retrieved subgraphs provide less typed structure beyond category and geographic proximity. The larger 20-option candidate set may also make the first-position decision more difficult. Averaged over both datasets, ASKG obtains the best MRR in all six dataset–backbone settings and the best ACC in five of six, with average gains of 5.9% and 8.3%, respectively. These results support an improvement in ranking the observed next POI, although the advantage is not uniform across all cutoff metrics.

Paired bootstrap analysis. Table 5 retains the draft's paired bootstrap values. A paired analysis jointly resamples the same test instances for both methods; each difference is the ASKG metric value minus the K-RagRec value. The stated 10,000 resamples and two-sided 95% intervals concern test-instance uncertainty, not training randomness. K-RagRec is a fixed statistical reference, not the strongest baseline for every metric.

Table 4. Overall performance on Yelp (10-option) and all 1083 Foursquare test instances (20-option). The best result for each backbone is in bold.

Backbone	Method	ACC	R@3	R@5	N@3	N@5	N@10	MRR
(a) Yelp								
Qwen2.5-7B	KGText	0.2070	0.5010	0.7220	0.3751	0.4656	0.5567	0.4201
	KAPING	0.2110	0.5140	0.7300	0.3840	0.4725	0.5618	0.4270
	K-RagRec	0.2380	0.5810	0.7860	0.4321	0.5167	0.5883	0.4580
	ASKG	**0.2650**	**0.6500**	**0.8450**	**0.4824**	**0.5629**	**0.6149**	**0.4924**
LLaMA-3.1-8B	KGText	0.2060	0.5000	0.7000	0.3751	0.4587	0.5498	0.4151
	KAPING	0.2050	0.4960	0.7150	0.3701	0.4601	0.5520	0.4160
	K-RagRec	0.2260	0.5980	0.8010	0.4347	0.5195	0.5912	0.4572
	ASKG	**0.2500**	**0.6900**	**0.8950**	**0.5027**	**0.5862**	**0.6207**	**0.4987**
GLM-4-9B	KGText	0.1970	0.5230	**0.7380**	0.3819	0.4703	0.5573	0.4197
	KAPING	0.1940	0.5160	0.7310	0.3761	0.4640	0.5521	0.4140
	K-RagRec	0.2180	0.5520	0.7350	0.4088	0.4862	0.5689	0.4398
	ASKG	**0.2450**	**0.5750**	0.7300	**0.4342**	**0.4977**	**0.5860**	**0.4579**
(b) Foursquare								
Backbone	Method	ACC	R@3	R@5	N@3	N@5	N@10	MRR
Qwen2.5-7B	KGText	0.1348	0.4654	0.7350	0.3190	0.4300	0.5174	0.3678
	KAPING	0.1302	**0.4746**	**0.7645**	0.3248	0.4440	0.5178	0.3687
	K-RagRec	0.1773	0.4552	0.7479	0.3208	0.4521	0.5158	0.3750
	ASKG	**0.1847**	0.4598	0.7599	**0.3389**	**0.4604**	**0.5436**	**0.4014**
LLaMA-3.1-8B	KGText	0.1653	0.4598	**0.7950**	0.3308	0.4685	0.5399	0.3954
	KAPING	0.1450	0.4146	0.7498	0.3016	0.4382	0.5248	0.3766
	K-RagRec	**0.1994**	0.5245	0.7821	**0.3798**	0.4655	0.5317	0.3925
	ASKG	0.1902	**0.5254**	0.7747	0.3778	**0.4798**	**0.5580**	**0.4190**
Gemma-2-9B	KGText	0.1653	0.4848	0.7849	0.3479	0.4703	0.5451	0.4020
	KAPING	0.1902	**0.5402**	**0.8199**	0.3879	0.5024	0.5630	0.4253
	K-RagRec	0.2050	0.5005	0.7849	0.3789	0.4968	0.5712	0.4368
	ASKG	**0.2373**	0.5152	0.7839	**0.3996**	**0.5088**	**0.5745**	**0.4441**

Table 5. Paired bootstrap comparison of ASKG and K-RagRec. Each difference is the ASKG metric value minus the K-RagRec metric value; intervals are two-sided 95% confidence intervals from 10,000 paired resamples. The reference is K-RagRec in every setting, not necessarily the strongest baseline for every individual metric.

Backbone	**ACC**		**MRR**	
	Difference (95% CI)	***p***	**Difference (95% CI)**	***p***
Yelp				
Qwen2.5-7B	0.0270 [0.0141, 0.0398]	< 0.001	0.0344 [0.0259, 0.0428]	< 0.001
LLaMA-3.1-8B	0.0240 [0.0110, 0.0369]	0.001	0.0415 [0.0327, 0.0502]	< 0.001
GLM-4-9B	0.0270 [0.0138, 0.0401]	< 0.001	0.0181 [0.0099, 0.0262]	< 0.001
Foursquare				
Qwen2.5-7B	0.0074 [−0.0110, 0.0253]	0.425	0.0264 [0.0113, 0.0423]	< 0.001
LLaMA-3.1-8B	−0.0092 [−0.0234, 0.0079]	0.249	0.0265 [0.0119, 0.0417]	0.001
Gemma-2-9B	0.0323 [0.0150, 0.0524]	0.001	0.0073 [−0.0055, 0.0197]	0.256

On Foursquare, the MRR gain is statistically significant for Qwen2.5-7B and LLaMA-3.1-8B, but not for Gemma-2-9B. The ACC difference is significant only for Gemma-2-9B; the Qwen2.5-7B and LLaMA-3.1-8B ACC differences are not significant. In particular, the negative LLaMA ACC difference is consistent with its lower point estimate in Table 4. These

results support a ranking improvement on Foursquare without implying a uniform improvement in top-1 accuracy. All significance statements in Table 5 are pairwise statements relative to K-RagRec only; corresponding tests were not performed against KGText or KAPING, so the reported confidence intervals and p-values do not establish statistical superiority over every baseline. Elsewhere in this paper, "strongest baseline" denotes the best-performing baseline by point estimate for a given setting and metric, not a baseline confirmed significant in every pairwise test.

Training-seed and candidate-construction variation. Training seeds control stochastic model initialization and data order; candidate-construction seeds control the user subset and sampled option pools. We use five distinct training seeds, 9, 39, 5, 25, and 17, while holding the candidate sets fixed so that ASKG and K-RagRec are compared on the same instances in every run. This is a paired training-variability study, not repeated training with one unchanged seed and not a study of candidate-construction variability. Three sources of randomness are distinguished throughout this paper: training-seed variation, studied here with the candidate pool fixed; candidate-construction variation, which this study does not vary; and candidate-label permutation under a fixed candidate pool, which is addressed separately in Section 5.1 and Section 5.11. Table 6 reports the resulting mean and standard deviation.

Table 6. Robustness of ASKG and K-RagRec over five independent training seeds (9, 39, 5, 25, and 17). Candidate pools are fixed across runs; values are means ± standard deviations.

Setting	K-RagRec		ASKG	
	MRR	**ACC**	**MRR**	**ACC**
Yelp, Qwen2.5-7B	0.4604 ± 0.0017	0.2436 ± 0.0055	0.4937 ± 0.0025	0.2626 ± 0.0042
Yelp, LLaMA-3.1-8B	0.4590 ± 0.0037	0.2286 ± 0.0033	0.4999 ± 0.0023	0.2524 ± 0.0032
Yelp, GLM-4-9B	0.4456 ± 0.0035	0.2226 ± 0.0058	0.4552 ± 0.0020	0.2390 ± 0.0037
Foursquare, Qwen2.5-7B	0.3806 ± 0.0094	0.1773 ± 0.0043	0.4051 ± 0.0093	0.1828 ± 0.0072
Foursquare, LLaMA-3.1-8B	0.4014 ± 0.0052	0.2048 ± 0.0041	0.4226 ± 0.0031	0.1974 ± 0.0079
Foursquare, Gemma-2-9B	0.4311 ± 0.0063	0.2037 ± 0.0045	0.4493 ± 0.0105	0.2366 ± 0.0057

ASKG has higher mean MRR in all six settings and higher mean ACC in five; the exception is Foursquare with LLaMA-3.1-8B. The small standard deviations show that the reported MRR pattern is stable across the five training seeds under fixed candidate pools. This evidence quantifies training randomness only; candidate-construction variability and confidence intervals over independent runs are distinct questions.

5.3. Ablation Study

This section answers RQ2. The ablation study follows the order in which evidence moves through ASKG. Replacing spatio-temporal retrieval with semantic nearest neighbors tests whether topical relevance alone can identify evidence for a feasible next visit. Replacing typed encoding and bottleneck compression with independent-node averaging removes the relations within that evidence. At the language interface, removing cross-attention leaves the graph prefix available but prevents the candidate prompt from consulting it explicitly. Finally, removing candidate-aligned discrimination leaves only the full-vocabulary generation loss. This variant tests whether learning the answer format is sufficient to distinguish valid POI labels. Figure 2a compares all four variants across the three Yelp backbones. We first conduct the component-level analysis on Yelp, whose 15 relation types provide heterogeneous evidence paths for isolating the contribution of each graph module, and then repeat it on the sparser Foursquare graph (Figure 2b).

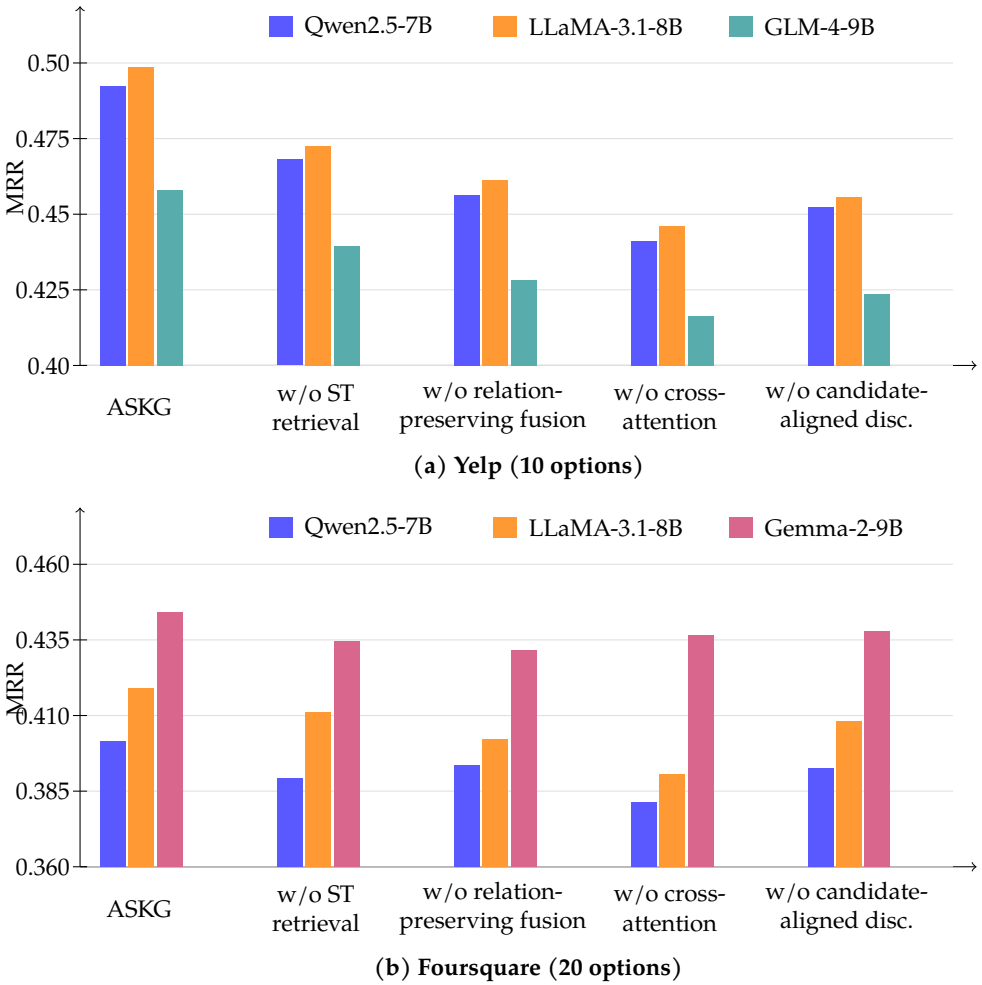


Figure 2. MRR of ASKG and four ablation variants under the candidate-constrained protocol: (**a**) Yelp with 10 options and (**b**) Foursquare with 20 options. The vertical axes begin at 0.40 and 0.36 for readability. All ablation results are measured experimentally; the ASKG values also appear in Table 4.

Every removal reduces MRR for all three backbones. Substituting semantic nearest neighbors for spatio-temporal retrieval costs 2.29 percentage points on average, showing that topical similarity cannot by itself account for mobility range and current intent. Replacing typed encoding and bottleneck compression with independent-node averaging produces a loss of 3.43 points, indicating that the connections among POIs, categories, and areas are not recovered by unordered pooling. The largest decrease, 4.85 points, occurs without cross-attention: merely placing a graph prefix before the prompt is insufficient. Removing candidate-aligned discrimination lowers MRR by 3.90 points because full-vocabulary language modeling compares the valid labels less directly. Across the variants, retrieval determines which evidence enters the model, while graph–language interaction and the loss determine how it affects the ranking.

Figure 2b applies the same four variants to Foursquare, whose graph has four relation types. Across the three Foursquare backbones, removing cross-attention causes the largest average MRR decrease (1.87 percentage points), followed by removing relation-preserving fusion (1.24 points), replacing spatio-temporal retrieval (0.99 points), and removing candidate-aligned discrimination (0.87 points). Cross-attention has the largest average effect in these reported Foursquare values, and every average reduction is smaller than on Yelp. This is an observed cross-dataset pattern, not evidence that relational sparsity alone caused the difference: candidate construction, interaction density, and backbone composition also change.

5.4. Cross-Backbone and Cross-Dataset Analysis

This section answers RQ3 using the backbone and dataset breakdown already reported in Table 4. Table 4 includes Qwen and LLaMA on both datasets and a third model family on each. ASKG has the highest MRR with every backbone, improving on the strongest baseline by 4.1–9.1% on Yelp and 1.7–7.0% on Foursquare. The magnitude remains backbone-dependent: the Gemma MRR gain is modest, ACC improves in five of the six settings, and the winning method at deeper cutoffs varies. The datasets also represent different relational conditions: Yelp contains 15 relation types and 640,156 triples, including category hierarchies, business-area links, and shared attributes, while Foursquare contains four relation types and 99,196 triples. Because interaction density, geographic coverage, candidate-set size, negative construction, and backbone composition also differ, this comparison establishes that grounding remains beneficial across both backbone families and relational densities but does not isolate the causal effect of graph richness alone. The component ablation on the sparser Foursquare graph is reported in Figure 2b, and a controlled variation of relational density within one dataset is given in Section 5.5.

Prior studies place ASKG within two complementary research directions. LLM4POI represents trajectories and venue descriptions through a language interface [10]; AgentMove organizes personal and urban context for agentic next-location prediction [11]; GA-LLM emphasizes geographic representation [12]; and MFC4POI models multiple factors of visit intent [13]. K-RagRec and KERAG-R instead emphasize retrieving external relational evidence for recommendation [18,19]. ASKG combines trajectory-conditioned evidence selection with relation-preserving compression and explicit interaction between candidate prompts and graph tokens. Its empirical pattern—higher MRR but mixed deeper-cutoff gains—suggests a benefit in ordering near the top of a supplied candidate set, rather than uniform improvement at every cutoff. This qualitative comparison identifies the method's research position without treating metrics from different datasets, candidate protocols, or backbones as a common leaderboard.

5.5. Relational Density and Graph Masking

The Yelp–Foursquare comparison changes interaction density, geographic coverage, candidate construction, and backbone composition together with graph structure; it cannot identify graph richness as the cause of the performance difference. Figure 3 retains two prior-draft analyses on Yelp with Qwen2.5-7B: (a) reducing the number of retained relation types from 15 to 10, 7, and 4, and (b) masking relations for 0%, 50%, and 100% of candidate POIs while retaining their text. Relation-type removal changes semantic coverage as well as the number of edges, so it is not a pure edge-density intervention. Candidate masking concerns missing evidence for supplied venues, not an unseen-venue test.

In the retained relation-removal values, ASKG decreases by 0.0201 MRR (0.4924 to 0.4723), K-RagRec by 0.0200 (0.4580 to 0.4380), and KGText by 0.0077 (0.4201 to 0.4124). The graph-based methods show similar decreases and a larger drop than KGText; these numbers do not show a meaningfully stronger relation-removal effect for ASKG than for K-RagRec. Their margin remains approximately 0.034. In the candidate-masking values, ASKG decreases to 0.4679, versus 0.4415 for K-RagRec and 0.4173 for KGText. Because candidate text remains accessible, the residual performance is consistent with the pretrained language pathway using supplied names, categories, attributes, and descriptions when graph evidence is unavailable. Subject to protocol verification, this describes graceful fallback under missing graph evidence. It does not establish the causal contribution of graph richness across datasets nor the ability to rank a genuinely unseen POI with no history or metadata.

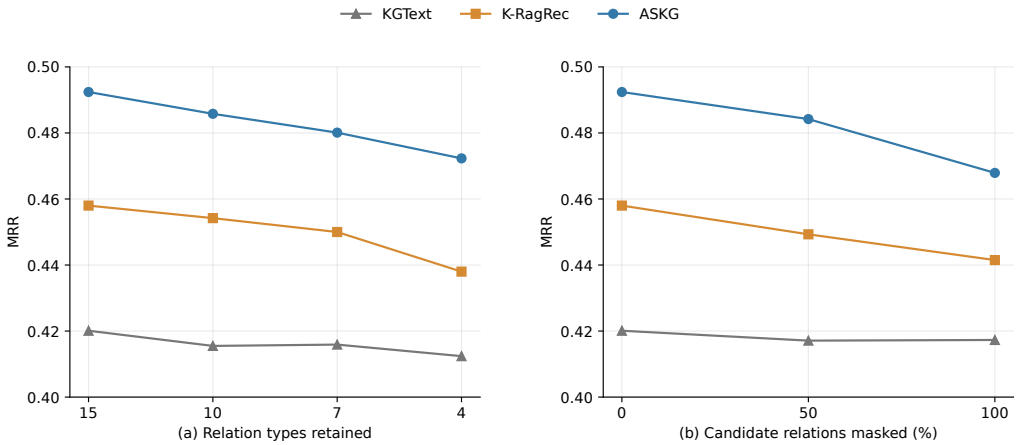


Figure 3. MRR on Yelp with Qwen2.5-7B (**a**) when relation types are progressively removed from the knowledge graph and (**b**) when the graph relations of a fraction of the candidate POIs are masked at inference.

5.6. Training Hyperparameter Sensitivity

This section answers RQ4 by examining the Stage 2 LoRA learning-rate ratio, which controls the relative update speed of the language adapter after the graph pathway is introduced. We sweep the ratio from 0.05 to 2.0 with the base learning rate fixed at 1×10^{-5}, train Stage 2 for four epochs, and keep all other settings fixed, with the resulting MRR reported in Figure 4. We use Gemma-2-9B on Foursquare as a controlled setting because this combination has the smallest MRR gain in the main comparison and therefore provides a stringent case for examining the optimization balance.

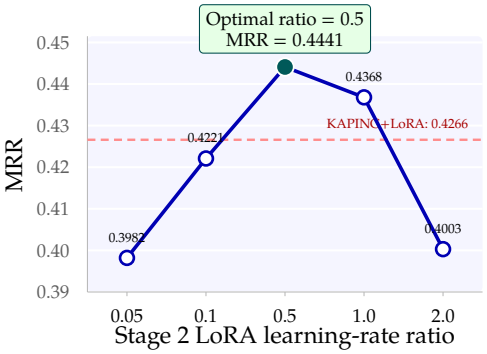


Figure 4. Stage 2 LoRA ratio sweep with a base learning rate of 1×10^{-5} and four Stage 2 epochs (Foursquare, Gemma-2-9B). MRR peaks at ratio 0.5; the dashed line marks the KAPING+LoRA baseline.

The sweep has a clear interior optimum in this controlled setting. Ratios below 0.5 update the LoRA adapter too slowly to accommodate the graph prefix introduced in Stage 2, whereas larger ratios increasingly disturb the language representation. The ratio of 0.5 achieves the highest MRR, 0.4441, exceeding the KAPING+LoRA reference of 0.4266. This result shows that balanced language-adapter updates are important after the graph pathway is introduced.

5.7. Retrieval Hyperparameter Sensitivity

Figure 5 reports three-point, one-parameter-at-a-time sweeps on Yelp with Qwen2.5-7B. The grids are geographic radius $\{3, 5, 8\}$ km, recall budget $\{8, 12, 16\}$, retained subgraphs $\{5, 10, 15\}$, and sequence decay $\{0.6, 0.8, 0.95\}$. The remaining settings are fixed. These local sweeps do not cover the mobility cap, geographic-weight bounds, interactions among

parameters, or the other five dataset–backbone settings. Without repeat-run uncertainty, small differences such as 0.0005 MRR do not establish an optimum.

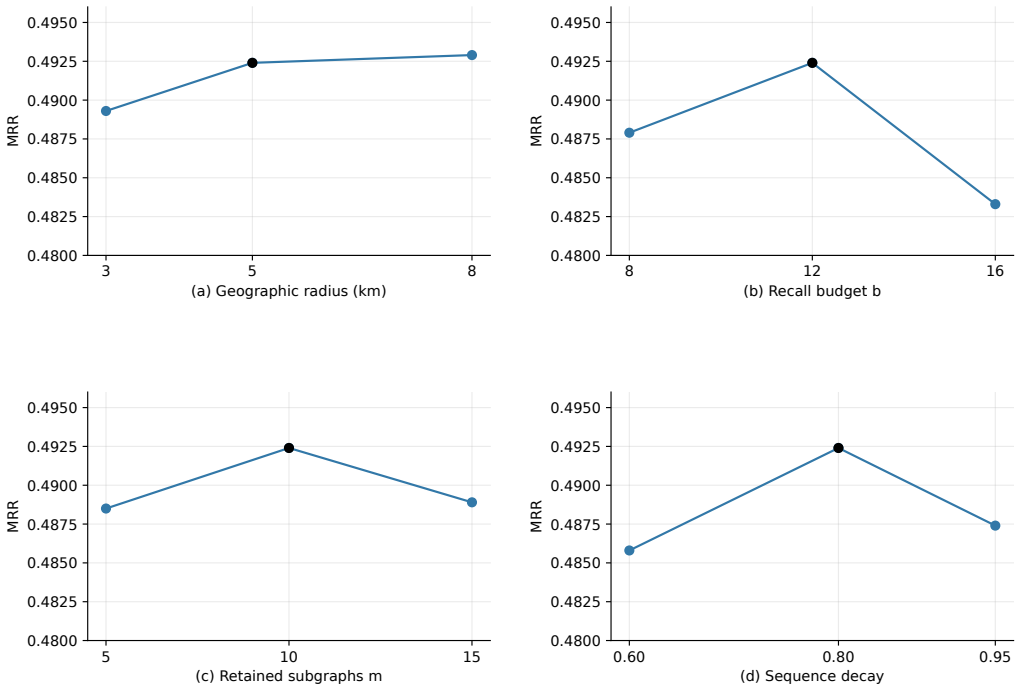


Figure 5. Sensitivity of MRR to the main retrieval hyperparameters on Yelp with Qwen2.5-7B, varying one parameter at a time. The black dot marks the reported setting.

MRR remains between 0.4833 and 0.4929 across the tested values. The reported configuration is close to the best point in each sweep, although an 8 km geographic radius is marginally higher than 5 km by 0.0005. The recall budget is the most sensitive parameter: increasing b from 12 to 16 lowers MRR from 0.4924 to 0.4833. The retained-subgraph count and sequence decay show smaller, smooth changes around their reported settings. These results indicate local robustness rather than insensitivity to all retrieval choices; the mobility cap and geographic-weight range are not included in the sweep.

5.8. Qualitative Case Analysis

Table 7 traces how retrieved evidence changes the probability assigned to the observed next POI. These selected cases illustrate spatial feasibility, preference recency, and relation preservation, together with one failure case; they do not independently validate a causal mechanism. Candidate names are replaced by functional descriptions to avoid exposing user-level location records.

Table 7. Representative next-POI cases. p^* is the candidate-normalized probability assigned to the observed next POI.

Case	Trajectory Signal	Evidence Emphasized by ASKG	K-RagRec p^*	ASKG p^*	Outcome
Spatial feasibility	Repeated weekday visits within a compact commuting area	Shared neighborhood and short geographic transition to a coffee venue	0.41	0.68	Correct rank moves from 2 to 1
Preference recency	Recent gym and health-food visits override older restaurant history	Sequence-position re-ranking favors health-related category links from recent POIs	0.36	0.61	Correct rank moves from 3 to 1

Table 7. *Cont.*

Case	Trajectory Signal	Evidence Emphasized by ASKG	K-RagRec p^*	ASKG p^*	Outcome
Relational bridge	Candidate descriptions use different surface terms	Two venues share a business area, parent category, and service attribute in the graph	0.44	0.64	Correct rank moves from 2 to 1
Sparse-evidence failure	Irregular trajectory with no stable activity area	Retrieved nodes mostly repeat category text already present in the candidates	0.54	0.52	Both methods rank the target second

The successful cases illustrate the intended roles of mobility constraints, recency, and shared relational attributes. Neighborhood evidence offers one explanation for the changed ranking in the spatial example; recent category links offer an explanation for the recency example; and shared attributes make the relational example easier to interpret. These selected examples do not identify which component caused a prediction change or establish a general mechanism. The failure example illustrates that repeating information already present in candidate text need not improve the target probability. We retain it to show a limitation of evidence grounding rather than treating the selected successes as systematic validation.

5.9. Efficiency and Resource Cost

The reported Yelp/Qwen2.5-7B configuration improves MRR from 0.4580 to 0.4924 relative to K-RagRec (+7.5%) while increasing latency from 496 to 538 ms (+8.5%). At batch size 1, peak GPU memory rises from 32.10 to 33.50 GB (+1.40 GB, or 4.4%). These numbers describe an effectiveness–resource trade-off for an LLM-based candidate ranker, not a claim that its full model is lightweight.

The timing protocol uses one NVIDIA A100 80GB GPU (NVIDIA Corporation, Santa Clara, CA, USA), batch size 1, 20 warm-up instances, and the mean over 1000 Yelp test instances. Total latency includes retrieval, graph processing, prompt encoding, and candidate-label logit computation. Figure 6 relates that latency to Table 4. These measurements quantify model-level feasibility; deployment capacity can then be calculated for a specified request rate, latency target, and serving topology.

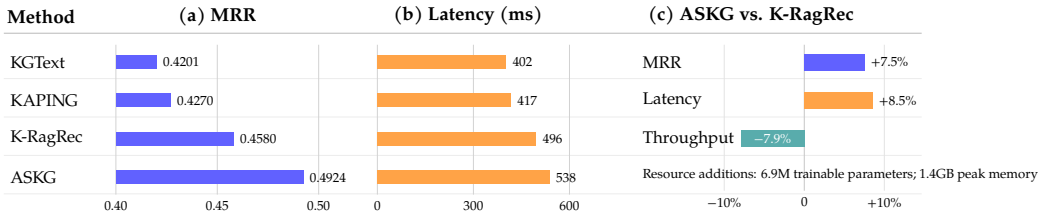


Figure 6. Effectiveness–efficiency comparison on Yelp with Qwen2.5-7B. Panels (**a**,**b**) report absolute MRR and mean measured ranking-pipeline latency, covering graph-evidence retrieval through option-label scoring and excluding upstream candidate generation, for all methods. Panel (**c**) compares ASKG directly with K-RagRec, the strongest baseline: ASKG improves MRR by 7.5%, increases latency by 8.5%, and reduces throughput by 7.9%.

In this representative configuration, ASKG increases total latency from 496 ms for K-RagRec to 538 ms per instance, while raising MRR from 0.4580 to 0.4924. It adds 6.9 M trainable parameters and 1.4 GB of peak memory relative to K-RagRec and reduces throughput from 2.02 to 1.86 samples/s. The additional cost comes mainly from adaptive dual-path retrieval, bottleneck compression, and query-conditioned graph–language interaction.

These measurements describe a single request. To approximate concurrent serving, Figure 7 reports throughput and peak memory for batch sizes 1 to 32 on the same GPU. The batched results provide model-level scaling evidence for subsequent capacity planning under a specified production workload.

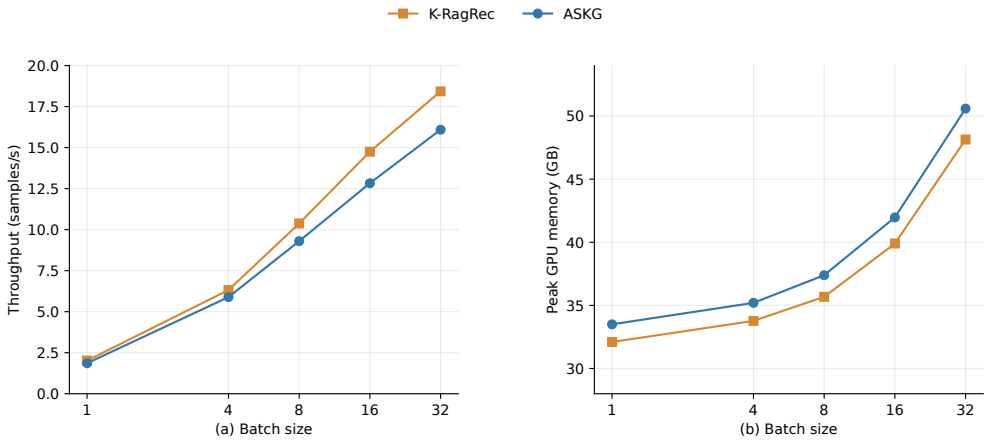


Figure 7. Throughput and peak GPU memory of K-RagRec and ASKG at different batch sizes on Yelp with Qwen2.5-7B.

Batching improves absolute throughput for both methods. From batch size 1 to 32, K-RagRec increases from 2.02 to 18.43 samples/s and ASKG from 1.86 to 16.09 samples/s. ASKG's throughput deficit grows from 7.9% to 12.7%, while its additional peak memory grows from 1.4 GB to 2.45 GB. The graph pathway therefore remains a measurable cost at larger batches, even though batching substantially improves total throughput.

5.10. General-Capability Check

The candidate-discrimination objective ($\alpha_{\text{loss}} = 0.05$ for the generation term) specializes the model for recommendation. Frozen backbone weights and low-rank adapters do not by themselves guarantee retention of general-purpose behavior. We therefore evaluate both the unmodified and ASKG-adapted Qwen2.5-7B and LLaMA-3.1-8B models under the same general-capability protocol.

The MMLU subset contains 200 zero-shot test questions sampled with seed 42:50 each from College Computer Science, High School Mathematics, High School Physics, and High School Biology. The generation probe contains 20 fixed prompts spanning general knowledge, logical reasoning, explanation, and text generation. Original and adapted models use identical prompts, greedy decoding, and a 256-token generation limit. Two annotators, blinded to model identity, independently score correctness, coherence, and relevance on a five-point scale; the reported generation score is the mean over annotators and prompts, with both the MMLU accuracy and the generation-probe score reported in Table 8 for the unmodified and ASKG-adapted backbones.

Table 8. General-capability comparison under the shared evaluation protocol. MMLU accuracy is measured on 200 zero-shot questions; the generation-probe score averages two blinded ratings over 20 prompts.

Backbone	Model	MMLU Subset (acc. %)	Gen. Probe (1–5)
Qwen2.5-7B	Unmodified	73.5	4.35
	ASKG-adapted	68.0	3.50
LLaMA-3.1-8B	Unmodified	66.5	4.20
	ASKG-adapted	62.0	3.30

Adaptation reduces MMLU accuracy by 5.5 percentage points for Qwen and 4.5 points for LLaMA, while the generation-probe score decreases by 0.85 and 0.90, respectively. We report this degradation directly as a consequence of task specialization and do not claim that the adapted models preserve full general-purpose capability.

5.11. Limitations

Task scale and recommendation paradigm. The benchmark supplies target-containing candidate sets; an upstream recall stage is a system-level positioning, not an evaluated component. The results do not test full-inventory recall, open-vocabulary generation, or simultaneous city-scale ranking. A bounded shortlist is the intended application interface. Comparing this LLM-based grounding pipeline with a conventional scorer would address the broader accuracy–resource choice between paradigms; the present results do not establish superiority over non-LLM recommenders.

Robustness and attribution. Training randomness and candidate-construction randomness are different sources of variation; a fixed-pool study addresses only the former and requires verified independent runs. Candidate-label permutation under a fixed candidate pool is a third, distinct source of variation and was not evaluated in the reported experiments. Removing relation types changes graph semantics as well as edge counts. Neither that limited within-Yelp study nor separately trained cross-dataset comparisons isolate graph richness as the cause of the Yelp–Foursquare difference. The local parameter sweeps do not establish robustness across all settings. Shared task inputs and LoRA adaptation also do not prove optimal baseline formatting or identical retained evidence.

Genuinely new venues. A venue that is new to one user need not be unseen during model training. Removing relations from an existing candidate retains its identity and textual description and therefore is not a test of a genuinely unseen POI. ASKG nevertheless retains the pretrained language pathway when graph evidence is unavailable. If a newly listed venue is supplied as a candidate with its name, category, location, attributes, or description, the LLM can use general semantic knowledge to interpret those features and compare them with the user's trajectory. This provides a fallback before interaction and graph evidence accumulate, but it does not give the model factual knowledge of a venue opened after pretraining unless current metadata are supplied. The fixed index and entity mapping also require an explicit insertion or out-of-index representation, which has not been tested. If both useful metadata and relational or interaction evidence are absent, the model lacks a venue-specific signal. Moreover, a ranking-only component cannot recover a venue omitted by upstream candidate generation. A direct cold-start test must withhold new venues and associated interactions from training and graph construction, introduce them after a time cutoff, and measure performance by metadata availability.

General capabilities and serving. Recommendation specialization measurably reduces performance on the reported MMLU subset and generation probe, positioning the adapted model as a specialized candidate ranker. Batched inference on one GPU provides fixed-hardware resource measurements. Translating these measurements into production capacity requires a target arrival rate, queueing and tail-latency objectives, retrieval and cache behavior, and a replication or sharding topology.

6. Conclusions

We study how external POI relations can support candidate-constrained next-visit generation. ASKG selects evidence according to mobility range and trajectory recency, preserves the retained relations within a fixed token budget, and exposes them to candidate representations. Candidate-aligned training maps this grounded representation to probabilities over the valid option labels.

ASKG achieves the highest MRR in all six dataset–backbone settings and the highest ACC in five. Relative to the strongest baseline for each backbone and metric, the average relative improvements are 5.9% in MRR and 8.3% in ACC. The MRR gains range from 4.1% to 9.1% on Yelp and from 1.7% to 7.0% on Foursquare. ACC, Recall, and NDCG vary more across datasets and backbones, with the most mixed results occurring on Foursquare. Overall, ASKG consistently improves MRR within the candidate-constrained generative setting, while gains at other ranking cutoffs depend on the dataset and backbone. These ranking gains come with measurable latency, throughput, and memory overhead, while task adaptation reduces performance on the reported general-capability evaluations; the adapted model should therefore be viewed as a specialized candidate ranker rather than a general-purpose LLM with preserved capabilities.

Future work will vary candidate-construction seeds and evaluate genuine cold start through a time-based split that withholds newly appearing venues from interaction-based graph construction and training then introduces them at test time under controlled levels of metadata availability. It will also examine production-scale serving, extend the task to open-vocabulary generation and substantially larger candidate sets, and study adaptive retrieval and fusion policies for graphs with different relational densities.

Author Contributions: Conceptualization, R.T. and D.C.; methodology, T.H. and Y.F.; software, J.Z.; validation, Y.F.; formal analysis, R.T. and T.H.; investigation, R.T. and J.Z.; data curation, T.H. and T.L.; writing—original draft preparation, R.T.; writing—review and editing, J.Z. and D.C.; visualization, T.L.; supervision, D.C.; funding acquisition, Y.F. and D.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Major Science and Technology Projects of Sichuan Province under grant no. 2024ZDZX0021 and the Major Program of the National Natural Science Foundation of China under grant no. T2293771.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets are obtainable from the Yelp Open Dataset, https://www.yelp.com/dataset (accessed on 25 May 2026) and the Foursquare NYC dataset, https://sites.google.com/site/yangdingqi/home/foursquare-dataset (accessed on 25 May 2026). The research code and reconstruction instructions are available at https://github.com/LittleMelon233/askg (accessed on 14 October 2026); users obtain backbone models from their original sources under the applicable terms. The current repository does not distribute fine-tuned checkpoints, so code availability is not equivalent to availability of a ready-to-evaluate trained model.

Conflicts of Interest: Authors Tianxiang He, Junlin Zhou, Yan Fu, and Duanbing Chen are employed by the company Chengdu Union Big Data Technology Incorporated. Authors Ruojia Tong and Tianyu Li declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Appendix A. Candidate-Label Tokenization

The saved tokenizer check uses `Qwen/Qwen2.5-7B`, `unsloth/Llama-3.1-8B-Instruct`, `THUDM/glm-4-9b-chat`, and `unsloth/gemma-2-9b-it`. The label strings are bare capital letters with no added punctuation or leading space. Table A1 lists every entry; each is a single token, and IDs are local to the respective vocabulary. The recorded assistant-marker checks reproduce the bare-label IDs, whereas leading-space variants differ. This mapping documents tokenization, not the absence of model label priors.

Table A1. Bare-label IDs in the saved tokenizer check. Yelp uses A–J and Foursquare uses A–T.

Label	Qwen2.5	LLaMA-3.1	GLM-4	Gemma-2
A	32	32	32	235280
B	33	33	33	235305
C	34	34	34	235288
D	35	35	35	235299
E	36	36	36	235291
F	37	37	37	235311
G	38	38	38	235319
H	39	39	39	235314
I	40	40	40	235285
J	41	41	41	235338
K	42	42	42	235333
L	43	43	43	235301
M	44	44	44	235296
N	45	45	45	235300
O	46	46	46	235302
P	47	47	47	235295
Q	48	48	48	235368
R	49	49	49	235294
S	50	50	50	235277
T	51	51	51	235279

References

1. Geng, S.; Liu, S.; Fu, Z.; Ge, Y.; Zhang, Y. Recommendation as language processing: A unified pretrain, personalized prompt and predict paradigm (P5). In Proceedings of the 16th ACM Conference on Recommender Systems, Seattle, WA, USA, 18–23 September 2022; pp. 299–315. [CrossRef]
2. Ngo, H.; Nguyen, D.Q. RecGPT: Generative pre-training for text-based recommendation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*; Association for Computational Linguistics: Kerrville, TX, USA, 2024; pp. 302–313. [CrossRef]
3. Liao, J.; Li, S.; Yang, Z.; Wu, J.; Yuan, Y.; Wang, X.; He, X. LLaRA: Large language-recommendation assistant. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington, DC, USA, 14–18 July 2024; pp. 1785–1795. [CrossRef]
4. Zhang, Q.; Yang, P.; Yu, J.; Wang, H.; He, X.; Yiu, S.-M.; Yin, H. A survey on point-of-interest recommendation: Models, architectures, and security. *IEEE Trans. Knowl. Data Eng.* **2025**, *37*, 3153–3172. [CrossRef]
5. Yang, D.; Fankhauser, B.; Rosso, P.; Cudré-Mauroux, P. Location prediction over sparse user mobility traces using RNNs: Flashback in hidden states. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, Yokohama, Japan, 7–15 January 2021; pp. 2184–2190. [CrossRef] [PubMed]
6. Sun, K.; Qian, T.; Chen, T.; Liang, Y.; Nguyen, Q.V.H.; Yin, H. Where to go next: Modeling long- and short-term user preferences for point-of-interest recommendation. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 214–221. [CrossRef]
7. Lian, D.; Wu, Y.; Ge, Y.; Xie, X.; Chen, E. Geography-aware sequential location recommendation. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Virtual Event, 23–27 August 2020; pp. 2009–2019. [CrossRef]
8. Luo, Y.; Liu, Q.; Liu, Z. STAN: Spatio-temporal attention network for next location recommendation. In Proceedings of the Web Conference 2021, Ljubljana, Slovenia, 19–23 April 2021; pp. 2177–2185. [CrossRef]
9. Wang, Z.; Zeng, J.; Zhong, L.; Liu, L.; Gao, M.; Wen, J. DSDRec: Next POI recommendation using deep semantic extraction and diffusion model. *Inf. Sci.* **2024**, *678*, 121004. [CrossRef]
10. Li, P.; de Rijke, M.; Xue, H.; Ao, S.; Song, Y.; Salim, F.D. 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, Washington, DC, USA, 14–18 July 2024; pp. 1463–1472. [CrossRef]
11. Feng, J.; Du, Y.; Zhao, J.; Li, Y. AgentMove: A large language model based agentic framework for zero-shot next location prediction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; Association for Computational Linguistics: Kerrville, TX, USA, 2025; pp. 1322–1338. [CrossRef]
12. Liu, Z.; Liu, W.; Zhu, H.; Yu, J.; Yin, J.; Lee, W.-C.; Wang, S. Geography-aware large language models for next POI recommendation. *arXiv* **2025**, arXiv:2505.13526. [CrossRef]

13. Song, Y.; Xu, W.; Liu, L.; Tiwari, P.; Tian, G.; Xie, Q.; Peng, M. MFC4POI: Multi-factor collaboration for next point-of-interest recommendation using large language models. *Inf. Process. Manag.* **2026**, *63*, 104824. [CrossRef]
14. Li, P.; Ao, S.; Xue, H.; Song, Y.; de Rijke, M.; Barthélemy, J.; Bednarz, T.; Salim, F.D. Refine-poi: Reinforcement fine-tuned large language models for next point-of-interest recommendation. *arXiv* **2025**, arXiv:2506.21599. [CrossRef]
15. Lv, D.; Ding, Q.; Xu, H.-D.; Sun, Z.; Wang, Z.; Xiong, F.; Xu, M. Reasoning over space: Enabling geographic reasoning for LLM-based generative next POI recommendation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Kerrville, TX, USA, 2026; pp. 7322–7336. [CrossRef]
16. Baek, J.; Aji, A.F.; Saffari, A. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. In *Proceedings of the Workshop on Natural Language Reasoning and Structured Explanations*; Association for Computational Linguistics: Kerrville, TX, USA, 2023; pp. 78–106. [CrossRef]
17. He, X.; Tian, Y.; Sun, Y.; Chawla, N.V.; Laurent, T.; LeCun, Y.; Bresson, X.; Hooi, B. G-Retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 132876–132907. [CrossRef]
18. Wang, S.; Fan, W.; Feng, Y.; Lin, S.; Ma, X.; Wang, S.; Yin, D. 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)*; Association for Computational Linguistics: Kerrville, TX, USA, 2025; pp. 27152–27168. [CrossRef]
19. Meng, Z.; Yi, Z.; Ounis, I. KERAG_R: Knowledge-enhanced retrieval-augmented generation for recommendation. *arXiv* **2025**, arXiv:2507.05863. [CrossRef]
20. Covington, P.; Adams, J.; Sargin, E. Deep neural networks for YouTube recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; pp. 191–198. [CrossRef]
21. Lee, S.; Choi, M.; Choi, E.; Kim, H.-Y.; Lee, J. GRAM: Generative recommendation via semantic-aware multi-granular late fusion. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Association for Computational Linguistics: Kerrville, TX, USA, 2025; pp. 33294–33312. [CrossRef]
22. Wang, D.; Huang, Y.; Gao, S.; Wang, Y.; Huang, C.; Shang, S. Generative next POI recommendation with semantic ID. *Proc. 31st ACM SIGKDD Conf. Knowl. Discov. Data Min.* **2025**, *2*, 2904–2914. [CrossRef]
23. Yin, F.; Liu, Y.; Shen, Z.; Chen, L.; Shang, S.; Han, P. Next POI recommendation with dynamic graph and explicit dependency. *Proc. AAAI Conf. Artif. Intell.* **2023**, *37*, 4827–4834. [CrossRef]
24. Huang, T.; Pan, X.; Cai, X.; Zhang, Y.; Yuan, X. Learning time slot preferences via mobility tree for next POI recommendation. *Proc. AAAI Conf. Artif. Intell.* **2024**, *38*, 8535–8543. [CrossRef]
25. Cui, Z.; Weng, Y.; Tang, X.; Lyu, F.; Liu, D.; He, X.; Ma, C. Comprehending knowledge graphs with large language models for recommender systems. In Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, Padua, Italy, 13–18 July 2025; pp. 1229–1239. [CrossRef]
26. Ding, Q.; Xu, H.-D.; Zhang, W.; Lv, D.; Xia, C.; Xiong, F.; Xu, M. Why users go there: World knowledge-augmented generative next POI recommendation. *arXiv* **2026**, arXiv:2605.11807. [CrossRef]
27. Shi, Y.; Huang, Z.; Feng, S.; Zhong, H.; Wang, W.; Sun, Y. Masked label prediction: Unified message passing model for semi-supervised classification. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, Montreal, QC, Canada, 19–27 August 2021; pp. 1548–1554. [CrossRef] [PubMed]
28. Li, J.; Li, D.; Savarese, S.; Hoi, S. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *Proc. 40th Int. Conf. Mach. Learn.* **2023**, *202*, 19730–19742.
29. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. LoRA: Low-rank adaptation of large language models. In Proceedings of the Tenth International Conference on Learning Representations, Online, 25–29 April 2022.
30. Yelp, Inc. Yelp Open Dataset. Available online: https://www.yelp.com/dataset (accessed on 25 May 2026).
31. Yang, D.; Zhang, D.; Zheng, V.W.; Yu, Z. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Trans. Syst. Man. Cybern. Syst.* **2015**, *45*, 129–142. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.