GUIDER: Uncertainty Guided Dynamic Re-ranking for Large Language Models Based Recommender Systems

Cai Xu, Xujing Wang, Ziyu Guan[*], Wei Zhao, Meng Yan

School of Computer Science and Technology, Xidian University, China
{cxu@, xjwong@stu., zyguan@, ywzhao@mail., mengyan@stu.} xidian.edu.cn

Abstract

Large Language Models (LLMs) are increasingly integral to recommendation systems, offering sophisticated language understanding and generation capabilities. However, their practical application is often hindered by challenges such as data sparsity, the generation of unreliable or hallucinated recommendations, and a general lack of transparency in their decision-making processes. Existing mitigation strategies frequently introduce significant complexity or computational overhead. To address these limitations, particularly the critical gap in quantifying the confidence of LLM-generated recommendations, we propose **GUIDER**: Uncertainty Guided Dynamic Re-ranking for Large Language Models based Recommender Systems. This new framework innovatively leverages the logits produced by LLMs as evidence for recommended items. By employing a Dirichlet distribution, GUIDER decomposes the total predictive uncertainty into distinct Data Uncertainty (DU), reflecting inherent data ambiguity, and Model Uncertainty (MU), indicating the model's own conviction. This principled decomposition, achieved with a single inference pass, enhances transparency and trustworthiness. Based on the quantified DU and MU levels, our system dynamically adapts its recommendation strategy—adjusting output diversity—through a four-quadrant analysis that tailors responses to specific uncertainty profiles. Extensive experiments conducted in zero-shot recommendation settings validate the effectiveness of our approach. GUIDER consistently outperforms existing methods in reliability-aware scenarios, demonstrably improving recommendation quality. This framework not only advances the practical deployment of LLM-based recommenders by making them more dependable but also provides a robust foundation for future research into uncertainty-aware generative systems.

Code — https://github.com/Wang-Xujing/GUIDER

Introduction

Recommender Systems (RS) enables users to match and obtain content that meets their needs from massive amounts of content. By prioritizing items likely to interest users, RS reduces search effort and enhances discovery. Existing RS methods (Sarwar et al. 2001) often struggle with several challenges, such as data sparsity (Guo 2012), cold-start problem (Lam et al. 2008), and limited personalization depth (Hu et al. 2017). These challenges have encouraged interest in leveraging Large Language Models (LLMs) for RS (Acharya, Singh, and Onoe 2023), which has the potential to improve recommendation quality and user experience significantly. A prominent and effective paradigm is to employ LLM as a sophisticated ranking agent within the recommendation pipeline (Qu et al. 2024; Yu et al. 2024; Zhai et al. 2024). This paradigm can leverage LLMs' advanced language understanding and reasoning capabilities to perform a fine-grained ordering of a candidate list generated by an initial retrieval stage.

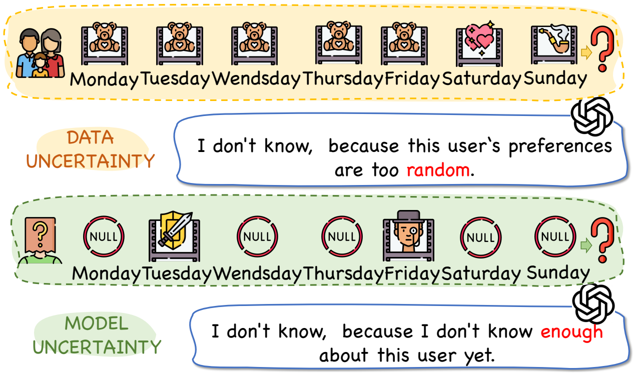


Figure 1: A conceptual illustration of the two primary sources of uncertainty in recommendation that our framework addresses. The scenario above depicts Data Uncertainty (DU), where a user's diverse and seemingly random interaction history makes it difficult to predict the next item with confidence. The scenario below illustrates Model Uncertainty (MU), where the model lacks sufficient evidence for a reliable prediction due to a sparse user history. Our work aims to quantify both types of uncertainty to produce more self-aware and trustworthy recommendations.

Despite their promise, LLM-based RS face an important challenge, i.e., hallucination (Huang et al. 2025). Existing strategies such as chain-of-thought (COT) prompting (Wei et al. 2022) and retrieval-augmented generation (RAG)

[*] Ziyu Guan is the corresponding author.

(Lewis et al. 2020), aim to enhance reasoning and reduce hallucinations. However, these methods often introduce additional complexity and computational overhead, while their effectiveness may vary with task complexity.

We posit that the issue of hallucination stems from two primary sources: the inherent ambiguity and randomness within the user data itself (e.g., a sparse interaction history), and the model's own lack of confidence or knowledge regarding a specific query. We contend that these root causes can be formally understood through the lens of predictive uncertainty. This total uncertainty can, in turn, be decomposed into two distinct components: Data Uncertainty (DU), which captures the ambiguity inherent to the data, and Model Uncertainty (MU), which reflects the model's own cognitive limitations. To address the hallucination problem from this perspective, our work proposes a new framework to empower the model with self-awareness by quantifying and decomposing these uncertainties, as illustrated in Figure 1. Our method efficiently leverages the logits produced by the LLM within a single inference pass, interpreting them as evidence within a Dirichlet distribution to derive the DU and MU values. This allows for a dynamic and low-overhead assessment of recommendation reliability, directly tackling the complexity issues while enhancing trustworthiness.

- We introduce a new perspective that frames LLM hallucinations in recommendation as a consequence of two quantifiable sources: Data Uncertainty from ambiguous user data, and Model Uncertainty from the model's internal knowledge gaps.
- We propose GUIDER, an efficient framework that models logits as evidence to decompose uncertainty into DU and MU. It then uses a dynamic, four-quadrant strategy to adapt the recommendation ranking, all within a single inference pass.
- Extensive experiments validate that GUIDER significantly outperforms strong baselines, confirming that our uncertainty-aware framework provides a meaningful foundation for building more trustworthy recommendation systems.

Related Work

Uncertainty in Recommendation Systems

Prior work in RS has addressed uncertainty by modeling user preference variability (Fan et al. 2021; Price and Messinger 2005; Wang et al. 2023; Xiong et al. 2024) or by calibrating output scores for tasks like exploration-exploitation trade-offs (Guo et al. 2017; Kweon et al. 2024; Kweon, Kang, and Yu 2022; Kweon and Yu 2024; Silva et al. 2023). However, these traditional methods are often designed for simple binary classification tasks, such as click-through rate prediction, and are not directly applicable to the complex, list-wise ranking scenarios handled by modern LLMs.

Uncertainty in Large Language Models

Uncertainty quantification in LLMs has primarily followed four methodological paths: **likelihood-based methods** that analyze token probabilities (Vazhentsev et al. 2023); **prompting techniques** that elicit confidence scores from the model (Kadavath et al. 2022; Xiong et al. 2023); **sampling approaches** that measure consistency across multiple outputs (Farquhar et al. 2024); and **training-based methods** that modify the model architecture itself, for example through Bayesian neural networks (Mielke et al. 2022). However, applying these general NLP methods to recommendation is challenging. The combinatorial nature of ranking makes likelihood estimation infeasible, verbalized confidence often proves unreliable for recommendation accuracy, and sampling-based approaches incur prohibitive computational costs.

A notable limitation of probability-based methods is the loss of evidence strength from raw logits during normalization, a flaw recently highlighted by (Ma et al. 2025a,b). This issue persists in recent works like (Kweon et al. 2025), which quantifies uncertainty using an entropy-based Plackett-Luce model; this approach still risks losing logit-level evidence and also requires multiple costly inferences. In contrast, our GUIDER framework directly models logits as evidence to decompose uncertainty (DU/MU) and enables a dynamic, single-pass adaptive strategy, making our approach both more direct in its quantification and more efficient in its application.

Preliminary

Problem Formulation Let $\mathcal{U}$ and $\mathcal{I}$ represent the sets of all users and items, respectively. For each user $u \in \mathcal{U}$, their interaction history is a chronologically ordered sequence $\mathcal{H}_u = [i_{u,1}, i_{u,2}, \ldots, i_{u,|\mathcal{H}_u|}]$. Each item $i \in \mathcal{I}$ is described by a textual representation t_i, such as its title and genres. The objective is to learn a function that, given a user's history $\mathcal{H}_u$, can accurately rank a set of candidate items $\mathcal{C}_u$ according to the user's preferences.

Candidate Set Construction To evaluate our model in a realistic re-ranking scenario, we construct a candidate set $\mathcal{C}_u$ for each user u in the test set. For each user, we first identify their ground-truth next item, i_{gt}, from their held-out interaction data. We then perform negative sampling by selecting $M-1$ items uniformly at random from the large set of items the user has never interacted with. The final candidate set is thus formed by the union $\mathcal{C}_u = \{i_{gt}\} \cup \{i_{neg_1}, \ldots, i_{neg_{M-1}}\}$. To mitigate any potential positional bias that might be present in LLMs, the order of items in this set is randomly shuffled before being incorporated into the prompt.

Prompt and Output Formulation Our interaction with the LLM is framed as an efficient list-wise ranking task. For each user u, a prompt $\mathcal{P}_u$ is constructed, containing their recent interaction history (e.g., the last 10 items) and the full, shuffled candidate set $\mathcal{C}_u$. Each item is represented by its title and genres to provide rich semantic context. Within the prompt, every candidate item is assigned a unique numerical index from 1 to M. The LLM is then instructed to identify the single most relevant item and respond with its corresponding index number.

Instead of parsing the generated text, we derive the full ranking from a single model inference. After the LLM

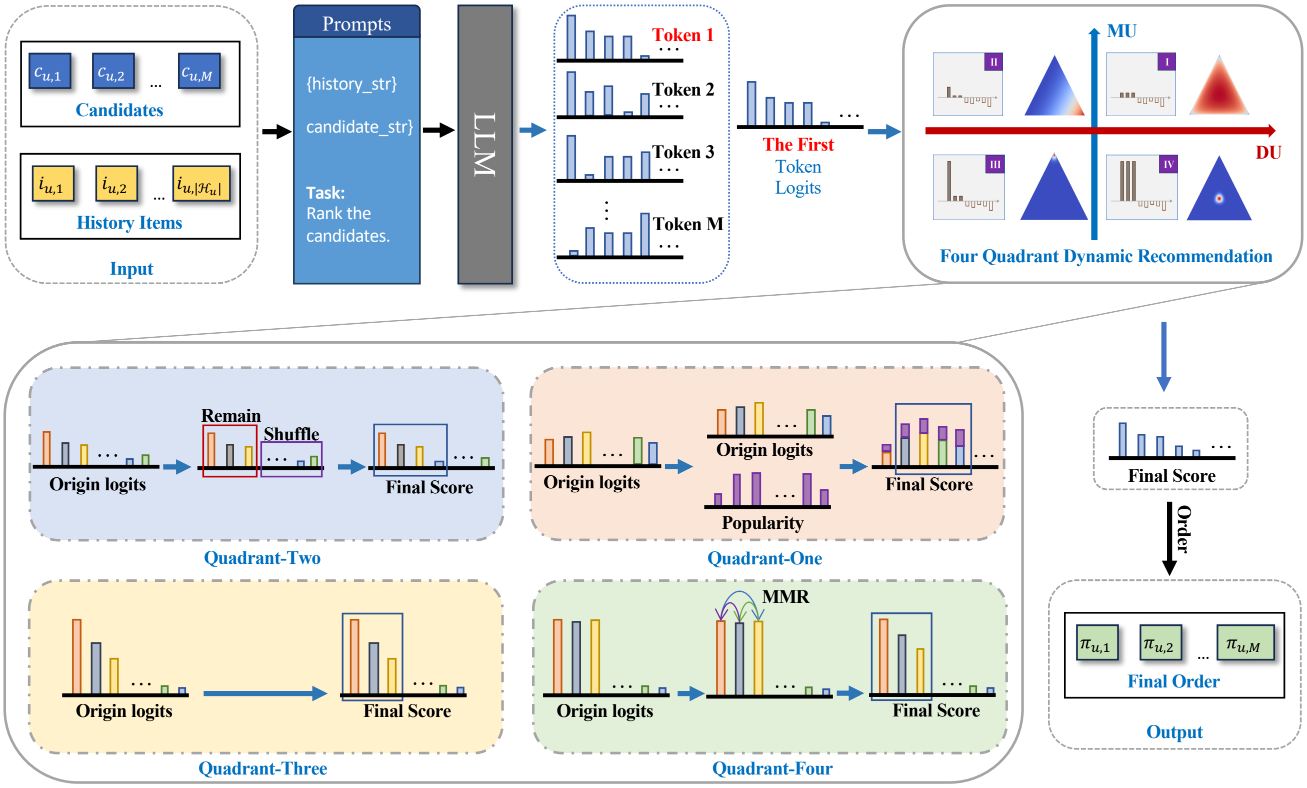


Figure 2: An overview of our proposed GUIDER framework for dynamic, uncertainty-aware recommendation. The process begins by constructing a prompt containing the user's history (H_u) and a set of candidate items (C_u). A single LLM inference then yields the initial logits for all candidates. These logits are used to compute Data Uncertainty (DU) and Model Uncertainty (MU), which categorize the recommendation into one of four quadrants. Each quadrant triggers a distinct strategy to produce the final scores: (I) blending with popularity for high ambiguity , (II) applying an exploration strategy (e.g., shuffling) for model uncertainty , (III) using the original logits directly in high-confidence scenarios , and (IV) employing MMR for diversity when the model is overconfident. The final ranked list (π_u) is then generated from the scores produced by the selected strategy.

processes the prompt, we extract the logits produced for the next-token prediction. Specifically, we retrieve the logit values for the tokens corresponding to our candidate indices ("1", "2", ..., "M"). This yields a score vector $\mathbf{z} = [z_1, z_2, \ldots, z_M]$, where z_k is the logit for the k-th candidate. The final ranked list π_u is obtained by sorting these scores in descending order. This efficient, one-shot process provides the basis for both our final ranking and our uncertainty analysis.

Methodology

This section details our proposed framework, GUIDER. We first outline the end-to-end pipeline, illustrating how we derive a full ranking and uncertainty scores from a single model inference. Subsequently, we delve into the core of our contribution: a new method for quantifying and decomposing the predictive uncertainty of LLM-generated recommendations into Data Uncertainty (DU) and Model Uncertainty (MU). Finally, we describe the dynamic, four-quadrant strategy that leverages these uncertainty metrics to adapt its ranking policy, aiming to enhance recommendation quality and trustworthiness.

Uncertainty Quantification and Decomposition

Despite the promise of Large Language Models (LLMs) in leveraging their advanced language understanding for more nuanced recommendations, their application in real-world systems is frequently hampered by issues of reliability. As highlighted in the introduction, challenges such as data sparsity and a tendency for model hallucination—where LLMs generate outputs not grounded in the provided context—can lead to untrustworthy or irrelevant suggestions. This lack of transparency and dependability is a significant barrier to their deployment. To address this critical gap, we introduce a framework to quantify and interpret this unreliability. Instead of merely accepting the generated output, our method probes the model's internal confidence by decomposing the total predictive uncertainty into its constituent components.

From Logits to Evidence: The Dirichlet Distribution Framework In the paradigm of Evidential Deep Learning (Sensoy, Kaplan, and Kandemir 2018; Xu et al. 2024), the outputs of a neural network (in our case, the LLM's logits corresponding to the candidate items) are utilized to parameterize a higher-order probability distribution. Our approach is inspired by (Ma et al. 2025a), which similarly demonstrates the effectiveness of using logits to decouple uncertainty. Specifically, we employ a Dirichlet distribution, which is a distribution over the parameters of a categorical (or multinomial) distribution.

Let z_k be the logit value generated by the LLM for the k-th candidate item out of K items in $\mathcal{C}_u$. We interpret these logits as raw evidence e_k for each item. Following established practices in EDL (Sensoy, Kaplan, and Kandemir 2018), the concentration parameters α_k of a Dirichlet distribution $\text{Dir}(\mathbf{p}|\alpha_1, \ldots, \alpha_K)$ are derived from this evidence. A common way to model this relationship is:

$$\alpha_k = \text{activation}(z_k) + c, \tag{1}$$

where $\text{activation}(z_k)$ is a non-negative function of the logit z_k and c is a constant, often 1, to ensure $\alpha_k > 0$. α_k appears to directly represent the evidence (logit value, possibly after ensuring positivity) for the k-th item.

The Dirichlet distribution is defined over the probability simplex, where $\mathbf{p} = (p_1, \ldots, p_K)$ are the probabilities of selecting each of the K items, such that $\sum p_k = 1$ and $p_k \geq 0$. The total strength of evidence, or the precision of the Dirichlet distribution, is given by $\alpha_0 = \sum_{k=1}^{K} \alpha_k$. A larger α_0 signifies a more concentrated (sharper) Dirichlet distribution, which in turn implies a more certain or confident prediction.

The probability density function (PDF) of the Dirichlet distribution is:

$$\text{Dir}(\mathbf{p}|\boldsymbol{\alpha}) = \frac{\Gamma(\alpha_0)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} p_k^{\alpha_k - 1}, \tag{2}$$

where $\Gamma(\cdot)$ is the gamma function. The expected probability for the k-th item under this Dirichlet distribution is:

$$E[p_k] = \frac{\alpha_k}{\alpha_0}. \tag{3}$$

Types and Formulations of Uncertainty We decompose the total predictive uncertainty into two fundamental types: **Data Uncertainty (DU))** : Data uncertainty reflects the inherent randomness, noise, or ambiguity present in the underlying data generating process. In the context of recommendations, this could arise from intrinsically unpredictable user preferences, conflicting interaction signals, or items that are inherently difficult to distinguish based on available information. This type of uncertainty is generally considered irreducible through model improvements alone if it is a true property of the data. Data Uncertainty is:

$$DU = -\sum_{k=1}^{K} \frac{\alpha_k}{\alpha_0} (\psi(\alpha_k + 1) - \psi(\alpha_0 + 1)), \tag{4}$$

where α_k is the evidence for the k-th candidate item, $\alpha_0 = \sum_{k=1}^{K} \alpha_k$, and $\psi(\cdot)$ denotes the digamma function, which is the logarithmic derivative of the gamma function $\psi(x) = \frac{d}{dx} \log \Gamma(x)$.

Mathematical Justification for DU: This formulation for DU is related to the expected entropy or variance measures associated with the Dirichlet distribution. The term $\frac{\alpha_k}{\alpha_0}$ is the expected probability $E[p_k]$. The digamma function $\psi(\cdot)$ appears in expressions for the differential entropy of the Dirichlet distribution and its moments. For instance, the variance of p_k under $\text{Dir}(\boldsymbol{\alpha})$ is $\text{Var}(p_k) = \frac{\alpha_k(\alpha_0 - \alpha_k)}{\alpha_0^2(\alpha_0 + 1)}$. High DU, as captured by the formula, generally arises when the evidence is spread among multiple candidates (high α_k for several k) or when there is conflicting evidence, leading to a Dirichlet distribution that is not sharply peaked around a single outcome. The specific form suggests it quantifies the spread or dispersion of beliefs attributable to the data's characteristics. This formula is consistent with advanced measures of uncertainty in evidential frameworks that consider the shape and spread of the evidence distribution.

Model Uncertainty (MU): Model uncertainty, on the other hand, captures the model's own lack of knowledge or confidence. This can stem from insufficient training data for certain types of inputs, limitations in the model's architecture or capacity to learn the underlying patterns, or encountering out-of-distribution samples. In principle, epistemic uncertainty can be reduced by exposing the model to more diverse and representative data or by improving the model itself. The Model Uncertainty is:

$$MU = \frac{K}{\sum_{k=1}^{K} (\alpha_k + 1)}, \tag{5}$$

where K is the number of top candidate items being considered, and α_k represents the evidence (logit value) for the k-th item.

Mathematical Justification for MU: This formulation of MU is directly analogous to the concept of vacuity or overall uncertainty mass (u) in Subjective Logic and some Evidential Deep Learning formulations. If we consider the Dirichlet parameters to be $\alpha'_k = \alpha_k + 1$ (where α_k is the raw evidence e_k), then the total strength becomes $\alpha'_0 = \sum_{k=1}^{K} (\alpha_k + 1) = (\sum_{k=1}^{K} \alpha_k) + K$. In this context, the uncertainty mass is often defined as $u = K/\alpha'_0$. Model Uncertainty is high when the total evidence accumulated by the model, $\sum \alpha_k$, is low. A low sum of evidence indicates that the model has not found strong support for any particular outcome, leading to a "flatter" or less concentrated Dirichlet distribution. This signifies a lack of conviction or knowledge on the part of the model. The formula $MU = K/((\sum \alpha_k) + K)$ captures this inverse relationship: as the total evidence $\sum \alpha_k$ increases, MU decreases, reflecting increased model certainty. This principled decomposition into DU and MU allows for a more nuanced understanding of the sources of unreliability in LLM-generated recommendations, which is critical for the dynamic adaptation strategy described next.

Dynamic Strategy via Four-Quadrant Uncertainty Analysis

Leveraging our uncertainty decomposition framework, we propose a dynamic recommendation strategy that adapts in real-time. The core principle of our strategy is to refine, not replace, the LLM's initial ranking. We treat the original logit scores as a strong signal and use the uncertainty quadrant to apply a tailored modification. Each recommendation scenario is categorized into one of four quadrants based on its DU and MU values, and a corresponding, principled approach is applied.

Quadrant I: High DU, High MU (High Ambiguity) This quadrant represents the most challenging scenario where the user's preferences are ambiguous (high DU) and the model is unconfident (high MU), making the raw logits highly unreliable. A classic example is a cold-start user with a sparse and thematically diverse history (e.g., a comedy, a documentary, and an action movie). In this high-risk scenario, the primary goal is to ensure robustness and provide a "safety net" recommendation. We achieve this through a popularity blending strategy, fusing the LLM's weak signal with the strong prior of global item popularity to compute a final score s'_k:

$$s'_k = (1 - \lambda_{pop}) \cdot \text{norm}(z_k) + \lambda_{pop} \cdot \text{norm}(\text{pop}_k), \quad (6)$$

where pop_k is the item's popularity score and λ_{pop} is a blending factor that shifts trust towards the "wisdom of the crowds" when the personalized signal is lost.

Quadrant II: Low DU, High MU (Model Uncertainty) This quadrant indicates that the user's preferences are consistent (low DU), but the model is unconfident (high MU), suggesting its high-level ranking is plausible but the fine-grained order is suboptimal. For instance, a user may have a clear interest in a niche genre that the model is unfamiliar with. The objective is therefore controlled exploration to discover a better local ranking within the high-potential candidate space identified by the model. We implement this with a hybrid exploration strategy, where the top $k_{exploit}$ items are chosen deterministically, and the rest are sampled from a shuffled pool of the next-best candidates:

$$\mathcal{S}_{explore} = \text{TopK}_{k_{explore}}(\text{Shuffle}(\mathcal{P}_{explore})) \quad (7)$$

where $\mathcal{P}_{explore}$ is a pool of the next M candidates. This introduces controlled randomness to correct for the model's uncertainty about the precise local ordering.

Quadrant III: Low DU, Low MU (High-Confidence) As the ideal scenario, this quadrant indicates clear user preferences (low DU) and a confident model (low MU), making the recommendation low-risk and reliable. For example, a user consistently watching a series where the next installment is in the candidate list. In this case, the goal is to maximize precision by fully trusting the model's well-grounded judgment. Therefore, we apply direct confidence-based ranking, generating the final list by sorting the original logits $\mathbf{z}$ without modification.

Quadrant IV: High DU, Low MU (Model Overconfidence) This quadrant signifies a risk of overconfidence, where the model is highly confident (low MU) despite ambiguous user preferences (high DU), potentially leading to a filter bubble. For example, the model might focus on only one of a user's multiple interests (e.g., action movies) while ignoring others (e.g., romantic comedies). To counteract this, our goal is to enhance diversity and ensure multiple facets of the user's preferences are represented. This is achieved using a diversity-aware re-ranking strategy based on Maximal Marginal Relevance (MMR), which iteratively selects items to balance relevance (logit score z_i) and dissimilarity from already selected items $\mathcal{S}$:

$$\arg\max_{c_i \in \mathcal{C} \setminus \mathcal{S}} \left[\lambda \cdot z_i - (1 - \lambda) \cdot \max_{c_j \in \mathcal{S}} \text{sim}(c_i, c_j) \right], \quad (8)$$

where $\text{sim}(c_i, c_j)$ is derived from the Jaccard similarity between the genres of items c_i and c_j and λ balances the trade-off between relevance and diversity.

Experiments

This section details the experimental evaluation of our proposed uncertainty-aware LLM-based recommendation framework, focusing on the zero-shot setting. Our experiments are designed to answer the following key research questions:

RQ1: How effectively do our proposed DU and MU metrics capture intuitive notions of uncertainty, such as the inherent ambiguity associated with users who have sparse interaction data?

RQ2: Does our proposed dynamic recommendation strategy, which adapts based on the four uncertainty quadrants, improve zero-shot recommendation performance compared to baseline approaches?

RQ3: How do various factors, such as user history length and candidate set size, influence the decomposed DU and MU values and the overall system performance in a zero-shot context?

We begin by describing the experimental setup, followed by detailed results and analyses corresponding to each research question, and conclude with overall performance, an ablation study, and complexity analysis.

Experimental Setup

Datasets We conduct experiments on three widely-used public datasets for sequential recommendation: MovieLens 10M (ML-10M) (Harper and Konstan 2015), Amazon Grocery and Gourmet Food (Amazon-Grocery)(Ni, Li, and McAuley 2019), and Steam(Kang and McAuley 2018). These datasets contain rich textual information suitable for LLM-based approaches. The statistics of these datasets are summarized in

Dataset	Interactions	Users	Items
ML-10M	10,000,054	71,567	10,681
Amazon-Grocery	5,074,160	2,695,974	287,209
Steam	7,793,069	2,567,538	15,474

Table 1: Dataset Statistics

Base Large Language Models We utilize three powerful, publicly available instruction-tuned LLMs as the backbone for our recommendation model in a zero-shot ranking setting: Llama3 (Grattafiori et al. 2024), Qwen2.5 (Yang et al. 2024), and Mistral (Jiang et al. 2023). These models are used directly with their pre-trained weights without any task-specific fine-tuning to strictly evaluate their zero-shot capabilities.

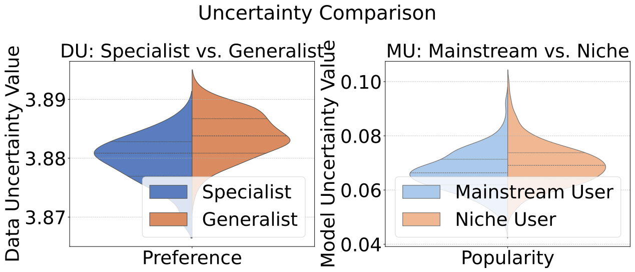


Figure 3: Split-violin plots validating DU and MU. (Left) Data Uncertainty (DU) is significantly higher for "Preference Generalists" (diverse history) than for "Niche Specialists" (consistent history). (Right) Model Uncertainty (MU) is significantly higher for "Niche Users" (preferring unpopular items) than for "Mainstream Users" (preferring popular items).

Implementation Details For our zero-shot experiments, prompts $\mathcal{P}_u$ are constructed for each user u, including a system message, user historical interactions $\mathcal{H}_u$ (up to $L_{max} = 20$), and a candidate item set $\mathcal{C}_u$ ($N_c = 100$) with textual features, and a ranking instruction. Candidate items for each user consist of the ground-truth test item and $N_c - 1$ randomly sampled negative items. Evidence α_k for each candidate item k is derived from LLM logits z_k using $\alpha_k = \text{softplus}(z_k) + 1$ for uncertainty calculation. Thresholds for High/Low DU and MU for the four-quadrant analysis are based on average values from the validation set. Recommendation quality is measured by NDCG@K (Järvelin and Kekäläinen 2002) and Recall@K (K=10, 20). All experiments are conducted using PyTorch on a single NVIDIA A100-80G GPU.

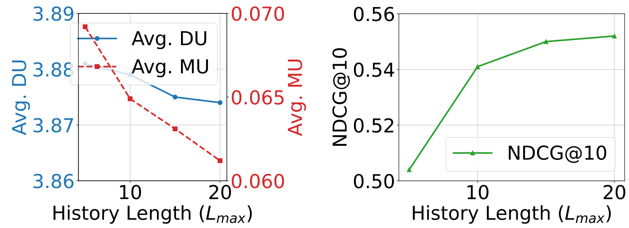
Compared Methods We evaluate our proposed method, GUIDER, against a comprehensive suite of baselines. These include traditional sequential recommenders like SASRec (Kang and McAuley 2018) and BERT4Rec (Sun et al. 2019), along with several recent LLM-based approaches such as PepRec (Yu et al. 2024), HSTU (Zhai et al. 2024), and RankGPT (Sun et al. 2023). Our primary comparison is against LLM4Rerank (Gao et al. 2025), which represents the current state-of-the-art (SOTA) for this task. Additionally, to perform a direct ablation study, we use the Standard LLM as a crucial internal baseline, which is our base LLM without the proposed dynamic uncertainty framework.

Effectiveness of Uncertainty Quantification (RQ1)

To answer RQ1, we validate our uncertainty metrics against intuitive user characteristics. First, we hypothesize that DU reflects preference ambiguity. We segment users by their preference consistency using the Gini impurity of genres in their history H_u, calculated as $Gini(H_u) = 1 - \sum_{g \in G}(p_g)^2$, where p_g is the proportion of genre g. As shown in the left panel of Figure 3, "Preference Generalists" exhibit significantly higher DU than "Niche Specialists", confirming DU quantifies preference diversity. Second, we hypothesize MU reflects the model's knowledge gap regarding niche items. We segment users by their average item popularity score, $AvgPop(H_u) = \frac{1}{|H_u|} \sum_{i \in H_u} Pop(i)$, where $Pop(i)$ is the global item popularity. The right panel of Figure 3 illustrates that "Niche Users" show significantly higher MU than "Mainstream Users". This validates that our metrics are strongly correlated with observable user patterns (preference consistency and item popularity), establishing a meaningful foundation for our framework.

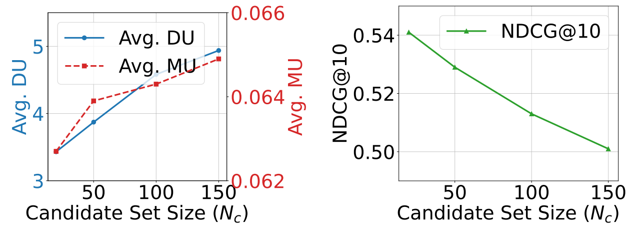
Performance of Dynamic Recommendation Strategy (RQ2)

To answer RQ2, we evaluate our proposed GUIDER framework against various baselines, with full results in Table 2. Our best variant, Ours(Qwen2.5), achieves state-of-the-art results on the majority of metrics. It significantly outperforms the Standard LLM baseline, highlighting the value of our dynamic uncertainty-aware strategy. Compared to the strong SOTA method LLM4Rerank, our model also shows superior performance overall, although LLM4Rerank remains competitive on select metrics on the Amazon-Grocery dataset. Finally, our framework proves robust across different backbones, with a consistent performance ranking of Ours(Qwen2.5) >Ours(Llama3) >Ours(Mistral), demonstrating the generalizability of our approach.



(a) DU and MU (b) NDCG@10

Figure 4: Impact of User History Length (L_{max}) on the Grocery dataset. (a) shows the effect on diversity (DU) and similarity (MU). (b) shows the effect on recommendation performance (NDCG@10).



(a) DU and MU (b) NDCG@10

Figure 5: Impact of Candidate Set Size (N_c) on the Grocery dataset. (a) shows the effect on diversity (DU) and similarity (MU). (b) shows the effect on recommendation performance (NDCG@10).

	ML-10M				Amazon-Grocery				Steam			
Model	N@10	R@10	N@20	R@20	N@10	R@10	N@20	R@20	N@10	R@10	N@20	R@20
SASRec	0.3487	0.4683	0.3759	0.5412	0.3208	0.4308	0.3458	0.4980	0.3801	0.5012	0.4097	0.5845
BERT4Rec	0.3172	0.4623	0.3426	0.5316	0.2918	0.4253	0.3152	0.4892	0.3458	0.4993	0.3734	0.5741
PepRec	0.4112	0.5584	0.4441	0.6477	0.3783	0.5137	0.4086	0.5959	0.4482	0.5975	0.4831	0.6931
HSTU	0.4106	0.4935	0.4434	0.5725	0.3778	0.4540	0.4079	0.5267	0.4476	0.5327	0.4827	0.6156
Standard LLM	0.3631	0.5031	0.3921	0.5836	0.3341	0.4628	0.3608	0.5369	0.3958	0.5413	0.4274	0.6299
RankGPT	0.4115	0.5586	0.4444	0.6479	0.3786	0.5139	0.4088	0.5961	0.4486	0.5977	0.4833	0.6934
LLM4Rerank	0.5941	0.6364	0.6416	0.7389	0.5466	0.6048	**0.5904**	0.7329	0.6476	0.6842	0.6994	0.7944
Ours(Mistral-7B)	0.5912	0.6815	0.6384	0.7837	0.5424	0.6252	0.5855	0.7190	0.6437	0.7288	0.6951	0.8381
Ours(Llama3-8B)	0.6088	0.6958	0.6575	0.8002	0.5585	0.6391	0.5891	0.7243	0.6639	0.7481	0.7168	0.8599
Ours(Qwen2.5-7B)	**0.6199**	**0.7086**	**0.6693**	**0.8149**	**0.5687**	**0.6510**	0.5899	**0.7372**	**0.6751**	**0.7616**	**0.7289**	**0.8752**

Table 2: Main performance comparison on NDCG@K and Recall@K (K=10, 20). The best result for each metric is in **bold**, and the second-best is underlined. Our proposed GUIDER consistently outperforms all baselines.

Method Variant	NDCG@20	Recall@20
GUIDER	**0.5899**	**0.7486**
w/o DU	0.4729	0.6135
w/o MU	0.4932	0.6218
w/o Dynamic Strategy	0.3608	0.5369

Table 3: Ablation study of our GUIDER framework. Performance is measured by NDCG@20 and Recall@20. The results confirm that each component—the dynamic strategy and the decomposition into DU and MU—is crucial for optimal performance. Note that 'w/o Dynamic Strategy' is equivalent to the 'Standard LLM' baseline.

Impact of Prompting Factors on Uncertainty (RQ3)

To answer RQ3, we analyze the impact of prompt factors. As shown in Figure 4, increasing history length (L_{max}) decreases both DU and MU while improving NDCG@10. This interaction highlights the complex trade-offs inherent in designing effective prompts for LLM-based rankers. Conversely, as shown in Figure 5, increasing the candidate set size (N_c) makes the ranking task more difficult, which is reflected by an increase in both DU and MU and a corresponding drop in NDCG@10. These results show our uncertainty metrics are sensitive to task configuration and difficulty.

Ablation Study

We conducted an ablation study to validate each component of our framework, with results in Table 3. The most significant performance drop occurs when removing the entire dynamic strategy (w/o Dynamic Strategy), confirming the substantial benefit of our adaptive approach over the static Standard LLM baseline. Furthermore, removing either the Data Uncertainty signal (w/o DU) or the Model Uncertainty signal (w/o MU) also significantly degrades performance, demonstrating that both uncertainty components are crucial for the strategy's effectiveness.

Method	Time Per Sample
SASRec	˜8ms
Standard LLM	˜530.9ms
LLM4Rerank	˜12339ms
GUIDER	**˜147.3ms**

Table 4: Complexity comparison of different methods. Our method maintains single-pass efficiency with minimal overhead.

Complexity Analysis

As a zero-shot framework, GUIDER involves no LLM fine-tuning, eliminating training costs beyond a single, efficient offline pass on a validation set to determine uncertainty thresholds. The framework's primary advantage lies in its inference efficiency. By requiring the LLM to predict only a single token and deriving the full ranking from the resulting logits, we avoid costly autoregressive generation. As shown in Table 4, this makes our GUIDER (˜147.3ms) significantly faster than both the Standard LLM (˜530.9ms) and the SOTA method LLM4Rerank (˜12339ms), providing a practical and scalable solution.

Conclusion

We introduce **GUIDER**, a new framework addressing unreliability in LLM-based recommenders. Its core innovation is leveraging logits as evidence to decompose predictive uncertainty into **Data Uncertainty (DU)** and **Model Uncertainty (MU)** in a single inference pass. Based on a four-quadrant analysis of these metrics, our framework dynamically adapts its strategy, delivering precise recommendations in low-uncertainty scenarios while strategically increasing diversity or using fallbacks when uncertainty is high. Extensive zero-shot experiments demonstrate that GUIDER significantly improves recommendation quality and its uncertainty metrics effectively signal reliability. Our work offers a robust foundation for developing more dependable and interpretable uncertainty-aware generative systems.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grants 62425605, 62133012, 62572375, and 62472340, in part by the Key Research and Development Program of Shaanxi under Grants 2025CY-YBXM-041, 2024GXYBXM-122, and 2022ZDLGY01-10, and in part by Natural Science Basic Research Program of Shaanxi (Program Nos. 2025JC-QYXQ-040, 2025JC-YBQN-900), and in part by Xidian University Specially Funded Project for Interdisciplinary Exploration (TZJHF202506).

References

Acharya, A.; Singh, B.; and Onoe, N. 2023. Llm based generation of item-description for recommendation system. In *Proceedings of the 17th ACM conference on recommender systems*, 1204–1207.

Fan, Z.; Liu, Z.; Wang, S.; Zheng, L.; and Yu, P. S. 2021. Modeling sequences as distributions with uncertainty for sequential recommendation. In *Proceedings of the 30th ACM international conference on information & knowledge management*, 3019–3023.

Farquhar, S.; Kossen, J.; Kuhn, L.; and Gal, Y. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017): 625–630.

Gao, J.; Chen, B.; Zhao, X.; Liu, W.; Li, X.; Wang, Y.; Wang, W.; Guo, H.; and Tang, R. 2025. Llm4rerank: Llm-based auto-reranking framework for recommendations. In *Proceedings of the ACM on Web Conference 2025*, 228–239.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; and et al. 2024. The Llama 3 Herd of Models. arXiv:2407.21783.

Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. Q. 2017. On calibration of modern neural networks. In *International conference on machine learning*, 1321–1330. PMLR.

Guo, G. 2012. Resolving data sparsity and cold start in recommender systems. In *User Modeling, Adaptation, and Personalization: 20th International Conference, UMAP 2012, Montreal, Canada, July 16-20, 2012. Proceedings 20*, 361–364. Springer.

Harper, F. M.; and Konstan, J. A. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.*, 5(4).

Hu, L.; Cao, L.; Wang, S.; Xu, G.; Cao, J.; and Gu, Z. 2017. Diversifying personalized recommendation with user-session context. In *IJCAI*, 1858–1864.

Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; et al. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2): 1–55.

Järvelin, K.; and Kekäläinen, J. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4): 422–446.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; Bressand, F.; and et al. 2023. Mistral 7B. arXiv:2310.06825.

Kadavath, S.; Conerly, T.; Askell, A.; Henighan, T.; Drain, D.; Perez, E.; Schiefer, N.; Hatfield-Dodds, Z.; DasSarma, N.; Tran-Johnson, E.; et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Kang, W.-C.; and McAuley, J. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, 197–206. IEEE.

Kweon, W.; Jang, S.; Kang, S.; and Yu, H. 2025. Uncertainty Quantification and Decomposition for LLM-based Recommendation. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, 4889–4901. Association for Computing Machinery.

Kweon, W.; Kang, S.; Jang, S.; and Yu, H. 2024. Top-Personalized-K Recommendation. In *Proceedings of the ACM Web Conference 2024*, 3388–3399.

Kweon, W.; Kang, S.; and Yu, H. 2022. Obtaining calibrated probabilities with personalized ranking models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 4083–4091.

Kweon, W.; and Yu, H. 2024. Doubly calibrated estimator for recommendation on data missing not at random. In *Proceedings of the ACM Web Conference 2024*, 3810–3820.

Lam, X. N.; Vu, T.; Le, T. D.; and Duong, A. D. 2008. Addressing cold-start problem in recommendation systems. In *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, 208–211.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474.

Ma, H.; Chen, J.; Wang, G.; and Zhang, C. 2025a. Estimating LLM Uncertainty with Logits. *arXiv preprint arXiv:2502.00290*.

Ma, H.; Pan, J.; Liu, J.; Chen, Y.; Zhou, J. T.; Wang, G.; Hu, Q.; Wu, H.; Zhang, C.; and Wang, H. 2025b. Semantic energy: Detecting llm hallucination beyond entropy. *arXiv preprint arXiv:2508.14496*.

Mielke, S. J.; Szlam, A.; Dinan, E.; and Boureau, Y.-L. 2022. Reducing conversational agents' overconfidence through linguistic calibration. *Transactions of the Association for Computational Linguistics*, 10: 857–872.

Ni, J.; Li, J.; and McAuley, J. 2019. Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 188–197. Association for Computational Linguistics.

Price, R.; and Messinger, P. R. 2005. Optimal recommendation sets: Covering uncertainty over user preferences. In *AAAI*, volume 10, 5.

Qu, H.; Fan, W.; Zhao, Z.; and Li, Q. 2024. Tokenrec: learning to tokenize id for llm-based generative recommendation. *arXiv preprint arXiv:2406.10450*.

Sarwar, B.; Karypis, G.; Konstan, J.; and Riedl, J. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, 285–295.

Sensoy, M.; Kaplan, L.; and Kandemir, M. 2018. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems*, 31.

Silva, N.; Silva, T.; Hott, H.; Ribeiro, Y.; Pereira, A.; and Rocha, L. 2023. Exploring Scenarios of Uncertainty about the Users' Preferences in Interactive Recommendation Systems. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1178–1187.

Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; and Jiang, P. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 1441–1450.

Sun, W.; Yan, L.; Ma, X.; Wang, S.; Ren, P.; Chen, Z.; Yin, D.; and Ren, Z. 2023. Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 14918–14937. Singapore: Association for Computational Linguistics.

Vazhentsev, A.; Tsvigun, A.; Vashurin, R.; Petrakov, S.; Vasilev, D.; Panov, M.; Panchenko, A.; and Shelmanov, A. 2023. Efficient out-of-domain detection for sequence to sequence models. In *Findings of the Association for Computational Linguistics: ACL 2023*, 1430–1454.

Wang, C.; Feng, F.; Zhang, Y.; Wang, Q.; Hu, X.; and He, X. 2023. Rethinking missing data: Aleatoric uncertainty-aware recommendation. *IEEE Transactions on Big Data*, 9(6): 1607–1619.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Xiong, M.; Hu, Z.; Lu, X.; Li, Y.; Fu, J.; He, J.; and Hooi, B. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Xiong, Y.; Liu, Y.; Qian, Y.; Jiang, Y.; Chai, Y.; and Ling, H. 2024. based recommendation under preference uncertainty: An asymmetric deep learning framework. *European Journal of Operational Research*, 316(3): 1044–1057.

Xu, C.; Si, J.; Guan, Z.; Zhao, W.; Wu, Y.; and Gao, X. 2024. Reliable conflictive multi-view learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 16129–16137.

Yang, A.; Yang, B.; Hui, B.; Zheng, B.; Yu, B.; Zhou, C.; Li, C.; Li, C.; Liu, D.; and et al. 2024. Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671*.

Yu, Y.; Qi, S.-a.; Li, B.; and Niu, D. 2024. PepRec: Progressive Enhancement of Prompting for Recommendation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 17941–17953.

Zhai, J.; Liao, L.; Liu, X.; Wang, Y.; Li, R.; Cao, X.; Gao, L.; Gong, Z.; Gu, F.; He, M.; et al. 2024. Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations. *arXiv preprint arXiv:2402.17152*.