Distilling LLM Reasoning into Dense Encoders: Bridging the Accuracy-Efficiency Gap in Recommendation

Donghee Han Daeyoung Roh A Young Kim Hwanjun Song[†] Mun Yong Yi[†]
KAIST, Republic of Korea
{handonghee, dybroh, ayoung, songhwanjun, munyi}@kaist.ac.kr
[†]Corresponding authors

Abstract

Large Language Models (LLMs) have shown remarkable potential in recommendation systems but suffer from prohibitive inference latency. Existing distillation approaches typically target Small Language Models (SLMs) or Conventional Recommendation Models (CRMs), yet face a critical trade-off between computational cost and semantic reasoning capacity. To bridge this accuracy-efficiency gap, we introduce Reasoning-to-Encoder Distillation (R2END), a framework that establishes a text encoder as the optimal student architecture for scalable recommendation. Unlike methods that mimic token generation, R2END compresses the teacher's reasoning into a dense vector space via a semantic alignment objective, effectively capturing user-item dynamics. Extensive experiments on four datasets demonstrate that R2END not only outperforms state-of-the-art baselines but also achieves drastically reduced latency, offering a sweet spot for recommendation.

1 Introduction

Large language models (LLMs) have emerged as a powerful paradigm for enhancing modern recommender systems (Bao et al., 2023; Yuan et al., 2023; Kim et al., 2025; Sheng et al., 2025). By leveraging their extensive world knowledge and sophisticated reasoning abilities, LLMs can move beyond traditional collaborative filtering (CF) to understand the nuanced, causal relationships behind user preferences. This has led to significant improvements in recommendation accuracy and diversity (Liu et al., 2025; Chen et al., 2025a; Han et al., 2025).

To make these powerful, yet computationally expensive, abilities practical for real-world applications, knowledge distillation has emerged as a prominent technique for transferring them to smaller, more efficient models (Gu et al., 2024; Panigrahi et al., 2025). This trend is also being

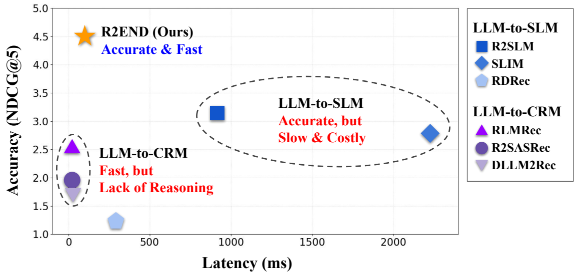


Figure 1: Accuracy-Latency trade-off in LLM distillation for recommendation.

adopted in the recommendation domain, where various studies are exploring methods to distill the nuanced reasoning of LLMs, aiming to build recommender systems that are both intelligent and scalable (Cui et al., 2024; Ren et al., 2024; Wang et al., 2024a,b). Specifically, these approaches focus on training the student to mimic the teacher's generative reasoning, striving to preserve the semantic depth of the decision-making process.

Despite their promise, existing distillation methods face a critical trade-off between reasoning depth and practical efficiency. As illustrated in Figure 1, current approaches are polarized into two directions. **LLM-to-SLM** approaches, while capable of generating text, suffer from persistent latency due to their autoregressive nature, making them unsuitable for real-time applications (Wang et al., 2024a,b). Furthermore, the reasoning generated by the teacher LLM often lacks fidelity or is misaligned, causing the student model to learn from flawed rationales.

Conversely, **LLM-to-CRM** methods prioritize inference speed but are insufficient to fully inherit the diverse abilities of LLMs (Cui et al., 2024; Ren et al., 2024). By simply training on recommendation results, these models fail to capture the internal reasoning process and lack the data and capacity required to incorporate the LLM's extensive parametric knowledge. Consequently, a solution that achieves the high fidelity of SLMs with the practical efficiency of CRMs remains an open challenge.

Findings of the Association for Computational Linguistics: ACL 2026, pages 22513–22531

To bridge this gap and address the aforementioned challenges, we propose **R**easoning-**to**-**En**coder **D**istillation for Recommendation (**R2END**), a novel framework designed to transfer aligned reasoning to an efficient encoder-based architecture. As illustrated in Figure 1, R2END serves as a sweet spot architecture that effectively resolves the Accuracy-Latency trade-off by capturing the semantic essence of reasoning without the computational overhead of autoregressive generation. Our core design choice is to distill the semantic representation of LLM's reasoning into a lightweight text encoder, which learns to produce a **reasoning-infused** user embedding that includes the semantics of the reasoning produced by the LLM without its inference overhead.

Furthermore, to maximize the effectiveness of this distillation, we address the issue of data quality. We observed that raw reasoning from LLMs can be factually unstable or misaligned with user behavior, and found that even large teachers (e.g. GPT-5.1 and Gemma3-27B) often fail to rank ground-truth items correctly (Less than 7% in Hit@10, Appendix E). To prevent the student from learning flawed rationales, we employ an oracle-guided reasoning process, compelling the teacher to provide high-fidelity reasoning strictly aligned with the ground truth. Our method utilizes ground truth in the training data as a privileged feature to reduce variance during distillation, as theoretically validated by a previous study (Yang et al., 2022).

Through extensive experiments on four real-world datasets, we demonstrate the effectiveness of our proposed method. R2END significantly outperforms state-of-the-art distillation-based recommenders in terms of accuracy and diversity. Crucially, by eliminating the need for an LLM at inference time, our approach drastically reduces latency and computational costs by orders of magnitude. These results validate that R2END offers a practical and effective pathway to building scalable recommendation systems that successfully incorporate the reasoning capabilities of LLMs.

Our contributions are summarized as follows:

- **Identification of Challenges:** We empirically identify the **Accuracy-Efficiency trade-off** in existing paradigms and uncover the issue of **reasoning misalignment**, demonstrating that LLM outputs often yield unreliable learning signals.
- **Novel Distillation Framework:** We propose Reasoning-to-Encoder Distillation (R2END), a novel architecture that effectively resolves the accuracy-efficiency trade-off. By distilling reasoning capabilities into a lightweight text encoder, we capture the semantic essence of LLM reasoning while eliminating the computational overhead of autoregressive inference.
- **Empirical Validation:** Through extensive experiments on real-world datasets, we validate that R2END achieves state-of-the-art performance and reduces inference latency, proving its practical viability for large-scale, real-time applications.

2 Related Work

2.1 LLM-based Recommendation

The integration of LLMs into recommender systems has opened new frontiers. Research in this area is primarily categorized into two main approaches: generative methods and embedding-based methods. Recent studies have further demonstrated that LLMs can enhance recommendation diversity, mitigate the cold-start problem (Kim et al., 2024; Liu et al., 2025, 2024), and provide greater explainability (Ramos et al., 2024), leading to a wide range of research aimed at leveraging these distinct advantages.

Early approaches leveraged the autoregressive capabilities of LLMs to directly generate recommendations. By reformulating the recommendation task as text generation, these methods treat item identifiers as tokens within a vocabulary and fine-tune an LLM to predict the next item (Geng et al., 2022; Bao et al., 2023; Lu et al., 2024; Kim et al., 2024). While these generative models have demonstrated impressive performance in capturing complex user preferences, their reliance on autoregressive generation results in significant inference latency, making them impractical for real-time applications.

To address the latency issue, another line of research utilizes LLMs as powerful feature encoders. In this approach, an LLM processes textual information associated with users or items (e.g., item descriptions, user reviews) to produce high-quality semantic embeddings (Liu et al., 2025; Sheng et al., 2025; Kim et al., 2025; Jia et al., 2025). Although this method is significantly faster at inference time, it treats the LLM as a static knowledge extractor.

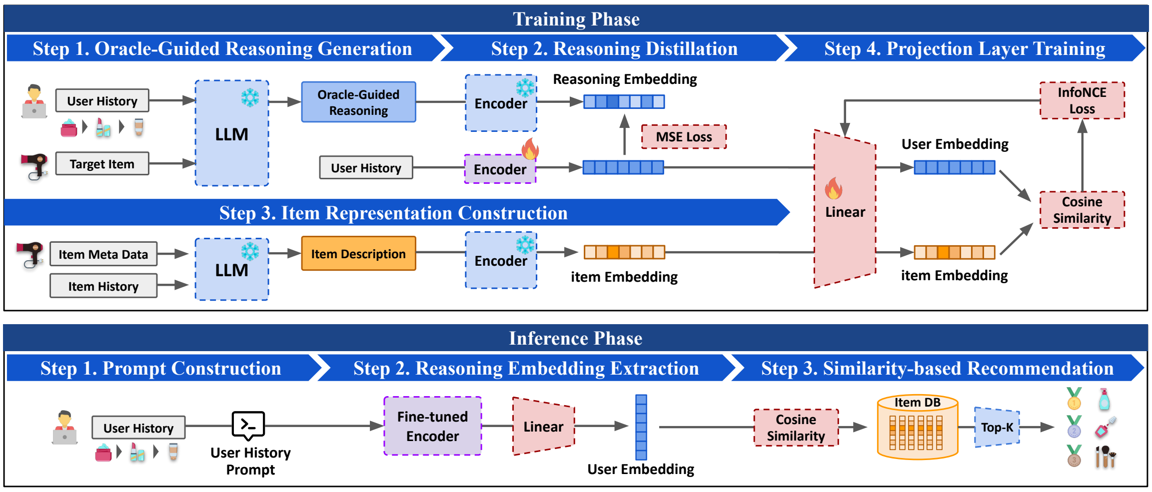


Figure 2: Overview of proposed method: **R2END**.

Consequently, it often fails to capture the dynamic, context-dependent reasoning that is a key advantage of LLMs, effectively using their knowledge but not their active reasoning process.

2.2 LLM Distillation for Recommendation

To capture the best of both the deep reasoning of LLMs and the efficiency of smaller models, knowledge distillation has emerged as a promising research direction (Gu et al., 2024; Panigrahi et al., 2025; Chen et al., 2025b). These methods aim to transfer the capabilities of a teacher LLM to a smaller, faster student model. Distilling into SLMs can transfer nuanced reasoning, but the resulting autoregressive students remain too slow for real-time applications and risk propagating the teacher's misalignments (Wang et al., 2024a,b). Conversely, distilling into traditional, non-generative recommenders achieves low latency but fails to capture the internal reasoning process, mimicking only the final outputs (Wang et al., 2025; Cui et al., 2024). In contrast to these approaches, our work focuses on distilling the LLM's reasoning capabilities into a pretrained text encoder, aiming to achieve both the reasoning of an LLM and the efficiency of an embedding-based system.

3 Reasoning-to-Encoder Distillation

In this section, we introduce our proposed method, **R**easoning-**to**-**En**coder **D**istillation for Recommendation (**R2END**). Our method is designed to distill the high-fidelity, aligned reasoning of an LLM into a lightweight and efficient text encoder for scalable recommendation. The overall architecture follows a teacher-student paradigm, consisting of three main stages: (1) Offline generation of oracle-guided reasoning and rich item descriptions using a powerful teacher LLM; (2) Training a student text encoder to mimic the LLM's reasoning process through distillation; and (3) An LLM-free inference stage that relies solely on the fast student encoder for recommendation. Figure 2 provides an overview of our framework.

3.1 Oracle-Guided Reasoning Generation

One of the challenges in distilling LLM reasoning is the risk of learning misaligned rationales generated by the teacher model. A critical observation in our study is that recent LLMs, despite their scale, struggle to rank ground-truth items correctly. To mitigate this, we introduce an oracle-guided generation process to ground the LLM's reasoning in factual user behavior, ensuring explicit alignment with the ground truth. For a given user u, let their chronological interaction history be denoted as a sequence of items $S_u = (i_1, i_2, \ldots, i_n)$. The ground-truth next item that the user interacts with is i_{n+1}. This sequence is then verbalized into a natural language sentence to form the user's raw history text, H_u. We construct a prompt, P_u, that includes both the user's history and this ground-truth item i_{n+1}, which serves as the "oracle."The LLM is then tasked with generating a reasoning text, R_u, that explains why user u would choose item i_{n+1} given their past actions. This process is formulated as:

$$R_u = \mathrm{LLM}(P_u(H_u, i_{n+1})) \tag{1}$$

By providing i_{n+1} as the oracle, we constrain the LLM's reasoning to be factually grounded, preventing it from generating speculative or incorrect rationales that diverge from the user's actual preferences. This ensures that the knowledge source for our distillation is of high quality and relevant to the recommendation task. At this stage, only the training set is used to prevent label leakage.

3.2 Reasoning Distillation into a Text Encoder

The core of our framework is to distill the reasoning capability, now captured in the aligned text R_u, into a computationally efficient student text encoder, Enc_S. The student encoder's goal is to learn to produce a "reasoning-infused"embedding directly from the user's raw history text, H_u. To create reasoning-infused target embeddings, we employ a pretrained text encoder (Enc) to convert the textual rationales, previously generated by our teacher LLM, into dense vector representations. To further enrich the supervised signal, we concatenate the generated reasoning R_u with the metadata of the ground-truth item, $M_{i_{n+1}}$ (e.g., title, category, brand). This combined text, $R'_u = R_u \oplus M_{i_{n+1}}$, is then encoded by the encoder to produce the target reasoning embedding, $e_u^R \in R^d$:

$$e_u^R = \text{Enc}(R'_u). \tag{2}$$

The student encoder Enc_S takes the user's history text as input and generates a corresponding user embedding, $e_u = \text{Enc}_S(H_u)$. We then train the student encoder with Mean Squared Error (MSE) between its output and the teacher's target embedding. The distillation loss is defined as:

$$\mathcal{L}_{\text{distill}} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \|e_u - e_u^R\|_2^2, \tag{3}$$

where $\mathcal{U}$ is the set of all users in the training data. This objective forces the student encoder Enc_S to internalize the semantic essence of the LLM's aligned reasoning process, enabling it to generate a user representation that simultaneously embeds both recommendation signals and the distilled reasoning.

3.3 Item Representation Construction

To obtain high-quality item embeddings that reside in the same semantic space as our user embeddings, we leverage LLMs to generate rich, context-aware item descriptions. Standard item metadata is often insufficient to provide the necessary context or describe salient features for recommendation. We therefore aim to generate a richer item representation grounded in both the extensive world knowledge of an LLM and the contextual behavior of users who have previously purchased the item. To overcome this, we prompt the LLM to create a descriptive text D_i for each item i, conditioning not only on its intrinsic metadata M_i but also on the interaction histories of users who previously purchased it. This provides valuable context about the item's key features and appeal. The generated description D_i is then encoded using the same pre-trained encoder Enc to produce the final item embedding $e_i \in R^d$:

$$D_i = \text{LLM}(P_i(M_i, H_i)) \tag{4}$$

$$e_i = \text{Enc}(D_i). \tag{5}$$

This ensures that both user and item embeddings are represented within a shared, meaningful semantic space, which is crucial for effective similarity-based recommendation.

3.4 Training Projection Layer

To further strengthen the supervisory signal and explicitly optimize for the recommendation task, we introduce a contrastive learning objective. The reasoning-infused user embedding e_u and the item embedding e_i are passed through a shared projection layer, $f(\cdot)$, to map them into a shared latent space for dense retrieval. This process yields the final user representation z_u and item representation z_i, which are optimized for the final similarity computation:

$$z_u = f(e_u), \;\; z_i = f(e_i). \tag{6}$$

We then employ the InfoNCE loss to maximize the similarity between a user and their ground-truth item (positive sample) while minimizing it for other items (negative samples). The contrastive loss is formulated as:

$$\mathcal{L}_{\text{InfoNCE}} = \sum_{u \in \mathcal{U}} - \log \frac{\exp(\cos(z_u, z_i^+)/\tau)}{\sum_{j \in \mathcal{I}_u^-} \exp(\cos(z_u, z_j)/\tau)}, \tag{7}$$

where $\mathcal{I}_u^-$ is the set of negative items of user u, z_i^+ is the representation of a positive item. $\cos(\cdot)$ is cosine similarity, and τ is a temperature hyperparameter.

To stabilize training and prevent overfitting, we incorporate an L_2-based regularization term applied to all embeddings involved in the loss:

$$\mathcal{L}_{\text{reg}} = \lambda \left(\|z_u\|_2 + \|z_i^+\|_2 + \frac{1}{N} \sum_{j=1}^{N} \|z_j\|_2 \right), \tag{8}$$

where λ is a regularization coefficient. The final training objective for the projection layer is a weighted sum of the contrastive loss and regularization term:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{InfoNCE}} + \mathcal{L}_{\text{reg}}. \quad (9)$$

3.5 LLM-Free Inference

One of the key advantages of our framework is its highly efficient and scalable LLM-free inference process, which is crucial for recommender systems. In contrast to prior works, we entirely exclude the LLM at inference time and instead rely solely on the student text encoder, which has been infused with the LLM's reasoning capabilities via our distillation process. For an incoming user request, their history text H_u is first converted into an embedding e_u by the encoder and then passed through a projection layer to produce the final user representation z_u. Recommendations are then produced by ranking all candidate items in the corpus $\mathcal{I}$ based on the cosine similarity between the user representation z_u and each pre-computed item representation z_i. The final set of top-K recommendations, $\mathcal{I}_K(u)$, is identified as follows:

$$\mathcal{I}_K(u) = \underset{i \in \mathcal{I}}{\arg \text{top-k}} \cos(z_u, z_i). \quad (10)$$

This architectural choice not only bypasses the significant latency and computational costs of autoregressive LLMs but also decouples the complex reasoning generation from the real-time serving loop, resulting in a highly scalable and practical system for deploying reasoning-based recommendations.

4 Experiment

In this section, we present extensive experiments to demonstrate the effectiveness of R2END, aiming to answer the following research questions (**RQs**).

- **RQ1** Does R2END achieve state-of-the-art performance compared to existing LLM-based and distillation-based recommendation baselines?
- **RQ2** How do the proposed Reasoning-to-Encoder architecture and the reasoning generation strategy contribute to overall performance improvement?
- **RQ3** How significant are the improvements in inference latency offered by R2END when compared to existing distillation-based approaches?

4.1 Experimental Setup

In this subsection, we describe the datasets, evaluation metrics, and baselines used in our experiments. In our experiments, we employ Gemma3 (Team et al., 2025) models of various scales: a 12B model serves as the teacher, a 1B model as the student, and a 4B model for the LLM-based recommendation baselines. A publicly available text encoder is utilized for both our proposed method and relevant baselines (Li and Li, 2024). More details are provided in the Appendix and our online repository[1].

Datasets. We conducted experiments on four widely used benchmark datasets: *Sports*, *Beauty*, *Toys* and *Yelp*. These datasets cover different domains, allowing us to evaluate the robustness and generalizability of our method across various scenarios.

Evaluation Metrics. We adopt standard ranking metrics, including Hit Rate (HR) and Normalized Discounted Cumulative Gain (NDCG), to evaluate recommendation performance. All evaluations are conducted over the full item pool, which consists of over 10K items per domain. This setup closely resembles real-world deployment scenarios and contrasts with prior LLM-based recommendation studies, which evaluate small-scale candidates.

Baselines. To evaluate our model, we compare it with diverse baselines in three categories.

(1) Conventional recommendation models. We include representative recommendation baselines such as GRU4Rec (Hidasi, 2015), BERT4Rec (Sun et al., 2019), SASRec (Kang and McAuley, 2018), FDSA (Zhang et al., 2019), and S^3-Rec (Zhou et al., 2020), which have been widely used for sequential recommendation.

(2) LLM-based recommendation methods. We include recent methods that utilize LLMs, such as AlphaRec (Sheng et al., 2025), LLMEmb (Liu et al., 2025), and LLM-SRec (Kim et al., 2025).

(3) LLM distillation-based recommendation methods. We compare our method against two categories of LLM distillation-based baselines: reasoning distillation methods such as SLIM (Wang et al., 2024b) and RDRec (Wang et al., 2024a), and hybrid approaches such as SLMRec (Xu et al., 2025), DLLM2Rec (Cui et al., 2024), and RLM-Rec (Ren et al., 2024) that leverage LLMs to enhance CRMs (e.g., SASRec).

[1] https://github.com/venzino-han/R2END

Table 1: Performance comparison of existing recommendation methods. The best results for each metric are highlighted in bold, and the second-best results are underlined.

Category	Method	LLM	Sports		Beauty		Toys		Yelp	
			H@5	N@5	H@5	N@5	H@5	N@5	H@5	N@5
Conventional Method	GRU4Rec	-	0.0129	0.0086	0.0164	0.0099	0.0097	0.0059	0.0152	0.0099
	SASRec	-	0.0233	0.0154	0.0387	0.0249	0.0463	0.0306	0.0223	0.0141
	BERT4Rec	-	0.0115	0.0075	0.0203	0.0124	0.0116	0.0071	0.0051	0.0033
	FDSA	-	0.0182	0.0122	0.0267	0.0163	0.0228	0.0140	0.0271	0.0170
	S^3-Rec	-	0.0251	0.0161	0.0387	0.0244	0.0443	0.0294	0.0168	0.0123
LLM-based Method	AlphaRec (MLP)	Gemma3 (4B)	0.0157	0.0099	0.0285	0.0183	0.0258	0.0174	0.0052	0.0024
	AlphaRec (LGCN)	Gemma3 (4B)	0.0210	0.0139	0.0280	0.0193	0.0107	0.0068	0.0044	0.0021
	LLMEmb	Gemma3 (4B)	0.0250	0.0160	0.0482	0.0310	0.0561	0.0369	0.0122	0.0076
	LLM-SRec	Gemma3 (4B)	0.0215	0.0101	0.0384	0.0237	0.0364	0.0225	0.0294	0.0173
	LEARN	Gemma3 (4B)	0.0115	0.0075	0.0157	0.0095	0.0213	0.0137	0.0047	0.0027
Teacher	SLIM(T)	Gemma3 (12B)	0.0273	0.0174	0.0452	0.0298	0.0524	0.0343	0.0491	0.0414
Student	SLIM(S)	Gemma3 (1B)	0.0247	0.0160	0.0419	0.0275	0.0499	0.0325	0.0486	0.0413
	RDRec	T5-Large (0.7B)	0.0045	0.0031	0.0162	0.0117	0.0053	0.0039	0.0114	0.0092
	SLMRec (8→4)	Gemma3 (1.9B)	0.0278	0.0162	0.0500	0.0308	0.0518	0.0321	0.0416	0.0303
	DLLM2Rec	-	0.0169	0.0104	0.0284	0.0174	0.0378	0.0248	0.0125	0.0080
	RLMRec	-	0.0302	0.0215	0.0357	0.0257	0.0141	0.0089	0.0133	0.0101
Ours	R2SLM (SFT)	Gemma3 (1B)	0.0226	0.0146	0.0464	0.0312	0.0529	0.0350	0.0518	0.0438
	R2SLM (Logit KD)	Gemma3 (1B)	0.0171	0.0113	0.0335	0.0218	0.0451	0.0289	0.0419	0.0332
	R2SASRec	-	0.0185	0.0115	0.0291	0.0181	0.0411	0.0264	0.0106	0.0066
	R2END	Text Encoder (0.3B)	**0.0344**	**0.0221**	**0.0664**	**0.0450**	**0.0712**	**0.0483**	**0.0595**	**0.0514**
Improvement			+13.91%	+2.79%	+37.76%	+45.16%	+26.92%	+30.89%	+21.18%	+24.15%

4.2 Overall Performance

To answer **RQ1** and validate the effectiveness of our proposed method, we compared our primary approach of distilling reasoning into an encoder (R2END) against state-of-the-art baselines and three distinct variants based on different student architectures. First, we implemented **R2SLM (SFT)**, which distills the same reasoning into an SLM using the generated reasoning text as the target for supervised fine-tuning. Second, to explore alternative distillation losses, we tested **R2SLM (Logit KD)**, which employs Logit-based Knowledge Distillation (Anshumann et al., 2025). Finally, we included **R2SASRec**, which attempts to transfer the reasoning capabilities into a conventional SASRec backbone.

Table 1 shows the overall recommendation performance. Our proposed method consistently outperforms all strong baselines, achieving an average performance improvement of 29% and up to 45% over the best baseline results. Furthermore, we observed that employing Logit KD for the SLM student yielded inferior performance compared to R2END. Similarly, R2SASRec also underperformed compared to our method, indicating that traditional recommendation backbones struggle to fully capture the nuances of textual reasoning. These results demonstrate that our proposed encoder-based student not only offers superior training stability and efficiency but also possesses sufficient semantic capacity to effectively internalize and utilize the complex reasoning data generated by the LLM.

These results validate the stability of embedding-based distillation compared to the volatility of token generation. While SLMs must navigate a brittle combinatorial space prone to "statistical shortcuts," R2END operates in a semantic vector space that encourages robust logic retention. This shift from token prediction to semantic alignment significantly enhances sample efficiency and generalization. Consistent with theoretical insights on the limitations of next-token prediction (Li et al., 2025), our framework demonstrates that distilling complex reasoning into a dense encoder provides a more structurally sound foundation for recommendation than autoregressive modeling.

Notably, our proposed method, R2END, surpasses the performance of the teacher model-based baseline (SLIM(T)). This result stems from the fundamental difference in how reasoning is utilized. During inference, the teacher model must generate reasoning without access to the ground-truth item, often leading to misaligned reasoning that diverges from actual user behaviors. In contrast, R2END is trained via our oracle-guided distillation, where it learns to map user history directly to a high-fidelity reasoning space derived from the ground-truth context. Consequently, the student encoder effectively filters out the noise and uncertainty inherent in the teacher's unguided generation, internalizing only the aligned causal logic required for accurate recommendation. This validates that our approach does not merely mimic the teacher but refines its reasoning capabilities into a more robust and "reasoning-infused"embedding space.

Table 2: Ablation study results. The best results for each metric are highlighted in bold.

Method	Sports				Beauty				Toys				Yelp			
	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10
R2END	**0.0344**	**0.0221**	**0.0517**	**0.0277**	**0.0664**	**0.0450**	**0.0958**	**0.0545**	**0.0712**	**0.0483**	**0.1035**	**0.0587**	**0.0595**	**0.0514**	**0.0701**	**0.0548**
(1) w.o. LLM Reasoning	0.0217	0.0135	0.0368	0.0183	0.0475	0.0309	0.0728	0.0390	0.0574	0.0377	0.0847	0.0464	0.0336	0.0251	0.0464	0.0292
(2) w.o. Text Encoder	0.0255	0.0163	0.0410	0.0213	0.0477	0.0307	0.0751	0.0395	0.0468	0.0308	0.0760	0.0402	0.0539	0.0452	0.0666	0.0493
(3) w.o. Oracle Guide (ALL)	0.0307	0.0198	0.0464	0.0248	0.0549	0.0366	0.0806	0.0449	0.0603	0.0402	0.0893	0.0496	0.0476	0.0398	0.0557	0.0425
(4) w.o. Oracle Guide (RS)	0.0313	0.0199	0.0469	0.0249	0.0573	0.0381	0.0828	0.0463	0.0605	0.0407	0.0896	0.0502	0.0288	0.0206	0.0412	0.0246
(5) w.o. Projection	0.0176	0.0116	0.0269	0.0146	0.0367	0.0235	0.0538	0.0290	0.0511	0.0341	0.0752	0.0419	0.0364	0.0284	0.0456	0.0313
(6) w.o. $\mathcal{L}_{reg}$	0.0320	0.0207	0.0506	0.0267	0.0616	0.0412	0.0914	0.0509	0.0671	0.0455	0.0989	0.0558	0.0579	0.0483	0.0699	0.0525

4.3 Ablation Study

To answer **RQ2** and analyze the contribution of each key component in our method, we conduct a comprehensive ablation study. We systematically remove or replace core components of our model and observe the impact on performance. The results are summarized in Table 2.

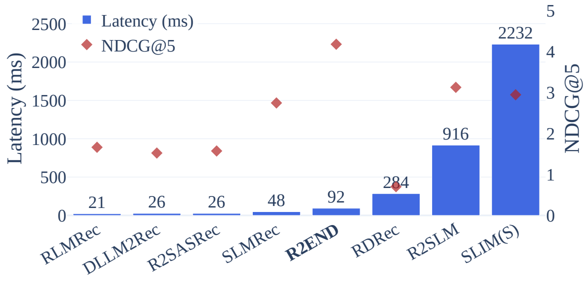


Figure 3: Comparison of latency and recommendation performance across different models.

4.3.1 Beyond Semantic Similarity: The Role of Distilled Reasoning.

We first investigate whether the student encoder learns rationale-aware mappings or merely semantic similarity to the target item. To verify this, we isolate the distilled reasoning signal in the variant **(1) w.o. LLM Reasoning**, which supervises the encoder solely with the ground-truth item's metadata embedding. Notably, this variant suffers the most severe performance degradation among all settings (Average 31.7% drop). This sharp decline provides pivotal empirical evidence: the ground-truth item alone serves only as the result ("what") and fails to provide sufficient supervisory signals for ranking. If the model were merely matching keywords or surface-level semantics, this variant should have performed comparably. However, the significant gap confirms that our method leverages LLM reasoning to explicitly verbalize the causal connectivity between user history and the target. This validates that the distilled reasoning acts as an indispensable semantic bridge, enabling the encoder to internalize the rationale ("why") behind user behaviors rather than simply memorizing item co-occurrences. Additionally, replacing the trainable encoder with a simple mean pooling operation (**(2) w.o. Text Encoder**) leads to significant drops, further proving that a sufficient model capacity is required to capture these complex dependencies.

4.3.2 Necessity of Alignment via Oracle Guidance.

Next, we investigate the role of our oracle-guided generation process. We compare our method with two variants: **(3) w.o. Oracle Guide (ALL)**, utilizing all unguided teacher outputs, and **(4) w.o. Oracle Guide (RS)**, a rejection sampling approach that selectively utilizes reasoning instances strictly aligned with actual user behaviors, adopting a filtering strategy similar to recent works (Tsai et al., 2024). Both variants significantly underperform our proposed method. Crucially, the failure of the unguided teacher demonstrates that raw LLM reasoning is prone to misalignment with domain-specific user behaviors. Furthermore, the inferior performance of rejection sampling indicates that merely filtering for correct predictions is insufficient; the process of reasoning itself must be explicitly grounded. These results counter the concern that oracle guidance is merely a shortcut; instead, they validate that the oracle serves as a vital calibration and denoising mechanism, ensuring the student learns from high-fidelity causal logic strictly aligned with actual user preferences, filtering out the noise inherent in open-ended generation.

4.3.3 Effect of Projection Layer and Regularization.

Finally, we examine the contribution of the final projection. We test our model **(5) w.o. Projection**, which removes the projection layer, and **(6) w.o. $\mathcal{L}_{reg}$**, which removes the regularization term. Removing the projection layer led to a noticeable performance decrease, while the removal of the regularization term also degraded performance, albeit to a lesser extent. This emphasizes the necessity of the projection layer and the regularization term for effectively learning a supervised signal tailored to the final recommendation task.

Table 3: Performance comparison of different student encoders.

Encoder	Sports				Beauty				Toys				Yelp			
	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10
Best Baseline	0.0278	0.0174	0.0433	0.0225	0.0482	0.0310	0.0767	0.0398	0.0561	0.0369	0.0838	0.0458	0.0491	0.0414	0.0588	0.0437
mxbai-embed-large (335M)	0.0344	0.0221	0.0517	0.0277	0.0664	0.0450	0.0958	0.0545	0.0712	0.0483	0.1035	0.0587	0.0595	0.0514	0.0701	0.0548
mxbai-embed-xsmall (24M)	0.0268	0.0176	0.0406	0.0220	0.0500	0.0324	0.0735	0.0399	0.0592	0.0396	0.0864	0.0484	0.0532	0.0474	0.0597	0.0495
bge-large-en (335M)	0.0303	0.0192	0.0473	0.0247	0.0512	0.0335	0.0780	0.0421	0.0658	0.0450	0.0959	0.0546	0.0575	0.0495	0.0674	0.0527
gte-base-en-v1.5 (100M)	0.0317	0.0215	0.0468	0.0263	0.0586	0.0402	0.0847	0.0486	0.0717	0.0502	0.0959	0.0580	0.0513	0.0429	0.0622	0.0464
Qwen3-embedding (0.6B)	0.0288	0.0184	0.0449	0.0236	0.0530	0.0348	0.0787	0.0431	0.0623	0.0418	0.0902	0.0508	0.0513	0.0431	0.0624	0.0466

4.4 Inference Efficiency

To answer **RQ3**, we investigate the trade-off between inference efficiency and recommendation effectiveness, as illustrated in Figure 3. In this figure, bars represent inference latency (left axis), while markers indicate the NDCG@5 performance (right axis). The results clearly demonstrate that our R2END strikes an optimal balance. While it is significantly more efficient than heavy baselines like R2SLM or SLIM(S), it maintains a distinct performance advantage. Although R2END exhibits higher latency compared to ultra-lightweight methods like SLMRec, this marginal cost is justified by a substantial gain in accuracy. This suggests that while a model compression-based approach (as seen in SLMRec) limits reasoning capabilities, our approach successfully retains the reasoning benefits of LLMs within a practical latency window suitable for real-world deployment.

4.5 Impact of Student Encoder Capacity and Generalizability

To investigate the generalizability and scalability of our framework, we conducted experiments using various text encoders as student models. The results, presented in Table 3, demonstrate that R2END consistently outperforms baselines across most configurations. Notably, we find that an encoder with just 100M parameters (e.g., GTE-base) is sufficient to surpass strong LLM-based baselines. While the ultra-lightweight 24M model showed performance degradation, indicating a capacity lower bound for capturing deep semantic reasoning, the 100M model effectively strikes the optimal balance between efficiency and accuracy. This finding is empirically significant; it confirms that R2END successfully distills the capabilities of massive teacher LLMs into highly efficient, compact encoders. This establishes a practical guideline for deploying state-of-the-art recommenders even in severely resource-constrained environments.

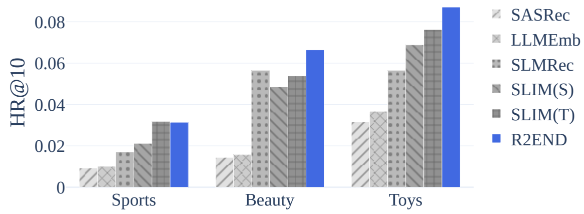


Figure 4: Hit@10 performance on long-tail items.

4.6 Long-tail performance

We investigate our model's capacity for improving recommendation diversity by evaluating its performance on long-tail items. Previous studies have shown that leveraging LLMs enhances recommendation diversity and particularly alleviates the long-tail problem (Liu et al., 2024, 2025; Han et al., 2025). We define the long-tail set as the least popular 80% of items. As illustrated in Figure 4, our proposed method, R2END, not only outperforms all baseline models but also surpasses the performance of the teacher model in this challenging setting. This result demonstrates that our distillation framework successfully inherits the LLM's renowned strength in recommending diverse and less popular items.

5 Conclusion

In this work, we addressed the critical trade-off between reasoning depth and inference latency in LLM-based recommendation. We introduced Reasoning-to-Encoder Distillation, a novel framework that successfully distills high-fidelity reasoning into a lightweight text encoder. Unlike previous approaches that mimic token generation, our core innovation lies in compressing the teacher's reasoning into a dense vector space. As demonstrated through extensive experiments, R2END not only achieves state-of-the-art performance but also drastically reduces inference latency. We hope our work inspires further research into semantic compression of reasoning, extending the utility of LLMs to other latency-sensitive applications. Ultimately, R2END offers a practical and effective pathway to building scalable recommender systems.

6 Limitations

While R2END demonstrates significant improvements in both accuracy and efficiency, we identify several limitations and discuss the broader implications of our work. First, regarding technical constraints, our reliance on a pre-trained text encoder as the student model introduces a dependency on model capacity. Our ablation study indicates that extremely lightweight models (e.g., 24M parameters) struggle to effectively internalize the complex reasoning distilled from the teacher LLM, suggesting a minimum threshold of semantic understanding is requisite. Second, although our method drastically reduces latency compared to generative LLM-based approaches, it exhibits slightly higher latency than ultra-lightweight distillation methods utilizing smaller architectures. However, we consider this a justifiable trade-off for the substantial gains in recommendation accuracy. Third, performance is intrinsically linked to the quality of textual data; in domains where rich textual data are sparse, the efficacy of semantic reasoning distillation may be constrained compared to collaborative filtering.

7 Ethics Statement

Use of Scientific Artifacts Our research leveraged open-source tools including PyTorch (Paszke et al., 2019), alongside pre-trained language models such as Qwen3-Next and Gemma3 obtained via the Huggingface (Wolf et al., 2019) library. For experiments involving LLMs, we utilized OpenAI's API under their sharing and publication policy (OpenAI, 2022).

Use of AI Assistants We only used Google Gemini to provide a better expression and to refine the wording. Some of the code used in the experiment was written with the assistance of Copilot.

References

Anshumann, Mohd Abbas Zaidi, Akhil Kedia, Jinwoo Ahn, Taehwak Kwon, Kangwook Lee, Haejun Lee, and Joohyung Lee. 2025. Sparse logit sampling: Accelerating knowledge distillation in LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18085–18108, Vienna, Austria. Association for Computational Linguistics.

Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2023. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 1007–1014.

Jiaju Chen, Chongming Gao, Shuai Yuan, Shuchang Liu, Qingpeng Cai, and Peng Jiang. 2025a. Dlcrec: A novel approach for managing diversity in llm-based recommender systems. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, WSDM '25, page 857–865, New York, NY, USA. Association for Computing Machinery.

Xiao Chen, Changyi Ma, Wenqi Fan, Zhaoxiang Zhang, and Li Qing. 2025b. C2KD: Cross-layer and cross-head knowledge distillation for small language model-based recommendation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 17827–17838, Vienna, Austria. Association for Computational Linguistics.

Yu Cui, Feng Liu, Pengbo Wang, Bohao Wang, Heng Tang, Yi Wan, Jun Wang, and Jiawei Chen. 2024. Distillation matters: Empowering sequential recommenders to match the performance of large language models. In *Proceedings of the 18th ACM Conference on Recommender Systems*, RecSys '24, page 507–517, New York, NY, USA. Association for Computing Machinery.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM Conference on Recommender Systems*, RecSys '22, page 299–315, New York, NY, USA. Association for Computing Machinery.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2024. Minillm: Knowledge distillation of large language models. In *ICLR*.

Donghee Han, Hwanjun Song, and Mun Yong Yi. 2025. Rethinking llm-based recommendations: A query generation-based, training-free approach. *arXiv preprint arXiv:2504.11889*.

B Hidasi. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

Wenyue Hua, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. 2023. How to index item ids for recommendation foundation models. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, SIGIR-AP '23, page 195–204, New York, NY, USA. Association for Computing Machinery.

Jian Jia, Yipei Wang, Yan Li, Honggang Chen, Xuehan Bai, Zhaocheng Liu, Jian Liang, Quan Chen, Han Li, Peng Jiang, and Kun Gai. 2025. Learn: knowledge adaptation from large language model to recommendation for practical industrial application. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'25/IAAI'25/EAAI'25. AAAI Press.

Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE.

Sein Kim, Hongseok Kang, Seungyoon Choi, Donghyun Kim, Minchul Yang, and Chanyoung Park. 2024. Large language models meet collaborative filtering: An efficient all-round llm-based recommender system. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 1395–1406, New York, NY, USA. Association for Computing Machinery.

Sein Kim, Hongseok Kang, Kibum Kim, Jiwan Kim, Donghyun Kim, Minchul Yang, Kwangjin Oh, Julian McAuley, and Chanyoung Park. 2025. Lost in sequence: Do large language models understand sequential recommendation? *Preprint*, arXiv:2502.13909.

Sunkyung Lee, Minjin Choi, Eunseong Choi, Hyeyoung Kim, and Jongwuk Lee. 2025. Gram: Generative recommendation via semantic-aware multi-granular late fusion. *arXiv preprint arXiv:2506.01673*.

Lei Li, Yongfeng Zhang, and Li Chen. 2023. Prompt distillation for efficient llm-based recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 1348–1357.

Xianming Li and Jing Li. 2024. AoE: Angle-optimized embeddings for semantic textual similarity. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1825–1839, Bangkok, Thailand. Association for Computational Linguistics.

Zhihao Li, Xue Jiang, Liyuan Liu, Xuelin Zhang, Hong Chen, and Feng Zheng. 2025. On the generalization ability of next-token-prediction pretraining. In *Forty-second International Conference on Machine Learning*.

Xinyu Lin, Wenjie Wang, Yongqi Li, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. 2024. Bridging items and language: A transition paradigm for large language model-based recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 1816–1826, New York, NY, USA. Association for Computing Machinery.

Qidong Liu, Xian Wu, Wanyu Wang, Yejing Wang, Yuanshao Zhu, Xiangyu Zhao, Feng Tian, and Yefeng Zheng. 2025. Llmemb: Large language model can be a good embedding generator for sequential recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12183–12191.

Qidong Liu, Xian Wu, Yejing Wang, Zijian Zhang, Feng Tian, Yefeng Zheng, and Xiangyu Zhao. 2024. Llm-esr: Large language models enhancement for long-tailed sequential recommendation. *Advances in Neural Information Processing Systems*, 37:26701–26727.

Wensheng Lu, Jianxun Lian, Wei Zhang, Guanghua Li, Mingyang Zhou, Hao Liao, and Xing Xie. 2024. Aligning large language models for controllable recommendations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8159–8172, Bangkok, Thailand. Association for Computational Linguistics.

OpenAI. 2022. Introducing chatgpt. `https://openai.com/blog/chatgpt`. Last accessed on 2024-01-15.

Abhishek Panigrahi, Bingbin Liu, Sadhika Malladi, Andrej Risteski, and Surbhi Goel. 2025. Progressive distillation induces an implicit curriculum. In *The Thirteenth International Conference on Learning Representations*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. *CoRR*, abs/1912.01703.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, et al. 2023. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36:10299–10315.

Jerome Ramos, Hossein A. Rahmani, Xi Wang, Xiao Fu, and Aldo Lipani. 2024. Transparent and scrutable recommendations using natural language user profiles. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13971–13984, Bangkok, Thailand. Association for Computational Linguistics.

Xubin Ren, Wei Wei, Lianghao Xia, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024. Representation learning with large language models for recommendation. In *Proceedings of the ACM Web Conference 2024*, WWW '24, page 3464–3475, New York, NY, USA. Association for Computing Machinery.

Leheng Sheng, An Zhang, Yi Zhang, Yuxin Chen, Xiang Wang, and Tat-Seng Chua. 2025. Language representations can be what recommenders need: Findings and potentials. In *ICLR*.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450.

Zhongxiang Sun, Zihua Si, Xiaoxue Zang, Kai Zheng, Yang Song, Xiao Zhang, and Jun Xu. 2024. Large language models enhanced collaborative filtering. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 2178–2188, New York, NY, USA. Association for Computing Machinery.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, and Alexandre Ramé et al. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Alicia Tsai, Adam Kraft, Long Jin, Chenwei Cai, Anahita Hosseini, Taibai Xu, Zemin Zhang, Lichan Hong, Ed H Chi, and Xinyang Yi. 2024. Leveraging llm reasoning enhances personalized recommender systems. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13176–13188.

Xinfeng Wang, Jin Cui, Yoshimi Suzuki, and Fumiyo Fukumoto. 2024a. RDRec: Rationale distillation for LLM-based recommendation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 65–74, Bangkok, Thailand. Association for Computational Linguistics.

Yue Wang, Dingyi Zhang, Haoyu Wenren, Yue Wang, and Yingming Li. 2025. Ekd4rec: Ensemble knowledge distillation from llm-based models to traditional sequential recommenders. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 1370–1374.

Yuling Wang, Changxin Tian, Binbin Hu, Yanhua Yu, Ziqi Liu, Zhiqiang Zhang, Jun Zhou, Liang Pang, and Xiao Wang. 2024b. Can small language models be good reasoners for sequential recommendation? In *Proceedings of the ACM Web Conference 2024*, WWW '24, page 3876–3887, New York, NY, USA. Association for Computing Machinery.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Wujiang Xu, Qitian Wu, Zujie Liang, Jiaojiao Han, Xuying Ning, Yunxiao Shi, Wenfang Lin, and Yongfeng Zhang. 2025. Slmrec: Distilling large language models into small for sequential recommendation. In *ICLR*.

Shuo Yang, Sujay Sanghavi, Holakou Rahmanian, Jan Bakus, and S.V.N. Vishwanathan. 2022. Toward understanding privileged features distillation in learning-to-rank. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Zheng Yuan, Fajie Yuan, Yu Song, Youhua Li, Junchen Fu, Fei Yang, Yunzhu Pan, and Yongxin Ni. 2023. Where to go next for recommender systems? id- vs. modality-based recommender models revisited. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 2639–2649, New York, NY, USA. Association for Computing Machinery.

Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, Xiaofang Zhou, et al. 2019. Feature-level deeper self-attention network for sequential recommendation. In *IJCAI*, pages 4320–4326.

Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 1893–1902.

A Datasets

Table 4: Statistics of the datasets.

Dataset	#Users	#Items	#Reviews	Density (%)
Sports	35,598	18,357	296,337	0.0453
Beauty	22,363	12,101	198,502	0.0734
Toys	19,412	11,924	167,597	0.0724
Yelp	30,431	20,033	316,354	0.0519

We conducted experiments on four widely used benchmark datasets : *Sports*, *Beauty*, *Toys*, and *Yelp*. These datasets cover different domains, allowing us to evaluate the robustness and generalizability of our method in diverse domains. Each dataset consists of user interactions, including a user ID, an item ID, a rating, a review, and a timestamp. The statistics of the dataset are provided in Table 4. These datasets are widely adopted by related studies (Geng et al., 2022; Rajput et al., 2023; Lee et al., 2025; Sun et al., 2024), which have been extensively explored over the past three years. We use five-core datasets, where both users and items have at least five interactions. For evaluation, we adopt the leave-one-out strategy, which is widely used in sequential recommendation research as a standard evaluation setup. This evaluation setup has been consistently adopted in previous studies (Kang and McAuley, 2018; Sun et al., 2019; Zhou et al., 2020; Li et al., 2023; Lee et al., 2025; Sun et al., 2024), ensuring comparability with existing methods.

B Baselines

To evaluate our method, we compare with a broad range of baseline models, grouped into three categories. Below, we briefly describe each baseline.

B.1 Conventional Recommendation Method

- **GRU4Rec** (Hidasi, 2015): One of the earliest sequential recommendation models, GRU4Rec employs gated recurrent units (GRUs) with a sequence-to-one pairwise ranking objective to capture temporal user behavior.

- **SASRec** (Kang and McAuley, 2018): Proposed to balance the efficiency of Markov Chains and the expressiveness of RNNs, SASRec uses self-attention to capture both short- and long-term user behavior. SASRec is designed to learn long-term user preferences based on only a small number of past actions by utilizing a self-attention mechanism. SASRec is a representative self-attention-based model.

- **BERT4Rec** (Sun et al., 2019): Designed to overcome the limitations of unidirectional models, BERT4Rec uses bidirectional self-attention to capture full sequence context. It employs a Cloze task to predict masked items, enabling richer sequence representations and improved performance across benchmarks.

- **FDSA** (Zhang et al., 2019): Aimed at capturing richer sequential patterns, FDSA models both item-level and feature-level transitions using separate self-attention blocks. By integrating heterogeneous item features and their dynamics, it improves recommendation performance over models that consider only item sequences.

- **S^3-Rec** (Zhou et al., 2020): To address data sparsity in sequential recommendation, S^3-Rec introduces self-supervised pre-training with four auxiliary objectives that capture correlations among items, attributes, and subsequences. By enhancing data representations through mutual information maximization, it achieves strong performance, especially under limited data scenarios.

B.2 LLM-based Recommendation Method

- **AlphaRec** (Sheng et al., 2025): AlphaRec is a recommendation framework that challenges the necessity of traditional ID-based embeddings. Its core finding is that the rich representation space of a LLM already implicitly contains collaborative signals. AlphaRec builds a simple yet effective recommendation model directly on top of item embeddings extracted from a language model, demonstrating that this approach can outperform leading ID-based methods. For our experiments, we evaluate both the MLP and the Light Graph Convolutional Network (LGCN) variants of AlphaRec to ensure a comprehensive comparison.

- **LLMEmb** (Liu et al., 2025): LLMEmb leverages LLMs to generate semantically rich item embeddings, addressing the long-tail problem in sequential recommendation. Through supervised contrastive fine-tuning and recommendation adaptation training, LLMEmb aligns LLM-generated embeddings with collaborative signals, leading to performance gains across various sequential recommendation system models.

- **LLM-SRec** (Kim et al., 2025): To address the limited sequential understanding of LLM-based recommenders, LLM-SRec input user and item representations from a pre-trained sequential recommendation model into an LLM. This method achieved high performance by integrating the semantic information from the LLM with the CF signal from the CF-based model.

- **LEARN** (Jia et al., 2025): LEARN is a framework designed to synergize open-world knowledge from pre-trained LLMs with collaborative signals, aiming to overcome the semantic limitations of traditional ID-based embeddings. To address computational complexity, it employs a frozen LLM as an item encoder within a twin-tower architecture, effectively aligning textual semantics with user-item interactions while preventing catastrophic forgetting.

B.3 LLM Distillation-based Recommendation Method

- **SLIM** (Wang et al., 2024b): This method proposes a LLM-based recommendation approach by distilling the reasoning capabilities of a large model into a smaller one. In our implementation, we utilize the Gemma3 12B model as the teacher and perform supervised fine-tuning on the Gemma3 1B student model using the generated reasoning text as the training data.

- **RDRec** (Wang et al., 2024a): This work addresses the problem that existing LLM-based recommenders do not explicitly learn the rationales behind user-item interactions. To solve this, a T5-based method where a smaller model learns by distilling rationales generated by a larger teacher LLM from user/item reviews. In our implementation, we utilize the Gemma3 12B model as the teacher and perform supervised fine-tuning on the T5-Large model as our student, using the generated reasoning text as the training data. We selected T5-Large as its parameter count is the most comparable to the other student models in our experiments. Furthermore, to mitigate the label leakage issue caused by token-level similarities in item IDs, as identified in recent studies (Lin et al., 2024), we adopted the sequential item ID assignment method from P5-SID for our evaluation (Hua et al., 2023).

- **SLMRec** (Xu et al., 2025): SLMRec proposes a method for faster and more efficient inference based on the empirical finding that many intermediate LLM layers are redundant for recommendation tasks. It uses knowledge distillation to transfer knowledge from a larger teacher model to a smaller student SLM. The architecture feeds embeddings from a traditional CF-based model into the LLM through an adapter. In our implementation, we designated 8 layers of a Gemma 12B model as the teacher and 4 layers as the student, incorporating item embeddings from SASRec. While this method is highly efficient, its reliance on CF-based embeddings and its failure to leverage the LLM's reasoning capabilities are significant limitations.

- **DLLM2Rec** (Cui et al., 2024): DLLM2Rec is a distillation strategy designed to transfer the capabilities of LLMs into lightweight conventional sequential models, thereby addressing inference latency constraints. To tackle challenges such as unreliable teacher knowledge and the capacity gap between models, it employs an importance-aware ranking distillation mechanism that filters and weights knowledge based on teacher confidence and student-teacher consistency. Additionally, it incorporates collaborative embedding distillation to integrate semantic knowledge from LLM embeddings with collaborative signals.

- **RLMRec** (Ren et al., 2024): RLMRec is a framework designed to enhance existing collaborative filtering models by integrating LLM-empowered representation learning. Addressing the limitations of ID-based recommenders and the noise inherent in implicit feedback, it incorporates auxiliary textual signals and employs an LLM-based profiling paradigm to capture complex user preferences.

Table 5: Performance comparison of existing recommendation methods (Top-10 metrics). The best results for each metric are highlighted in bold, and the second-best results are underlined. “H@10" and “N@10" denote Hit Rate and NDCG at rank 10, respectively.

Category	Method	LLM	Sports H@10	Sports N@10	Beauty H@10	Beauty N@10	Toys H@10	Toys N@10	Yelp H@10	Yelp N@10
Conventional Method	GRU4Rec	-	0.0204	0.0110	0.0283	0.0137	0.0176	0.0084	0.0263	0.0134
	SASRec	-	0.0350	0.0192	0.0605	0.0318	0.0675	0.0374	0.0390	0.0194
	BERT4Rec	-	0.0191	0.0099	0.0347	0.0170	0.0203	0.0099	0.0090	0.0045
	FDSA	-	0.0288	0.0156	0.0407	0.0208	0.0381	0.0189	0.0544	0.0293
	S^3-Rec	-	0.0385	0.0204	0.0647	0.0327	0.0700	0.0376	0.0341	0.0168
LLM-based Method	AlphaRec (MLP)	Gemma3 (4B)	0.0268	0.0135	0.0456	0.0238	0.0407	0.0221	0.0073	0.0034
	AlphaRec (LGCN)	Gemma3 (4B)	0.0328	0.0177	0.0432	0.0242	0.0179	0.0091	0.0070	0.0031
	LLMEmb	Gemma3 (4B)	0.0389	0.0205	0.0754	0.0398	0.0838	0.0458	0.0205	0.0103
	LLM-SRec	Gemma3 (4B)	0.0429	0.0171	0.0767	0.0358	0.0682	0.0327	0.0588	0.0267
	LEARN	Gemma3 (4B)	0.0188	0.0098	0.0286	0.0136	0.0349	0.0181	0.0096	0.0042
Teacher	SLIM(T)	Gemma3 (12B)	0.0433	0.0225	0.0688	0.0374	0.0786	0.0426	0.0561	0.0436
Student	SLIM(S)	Gemma3 (1B)	0.0403	0.0210	0.0664	0.0354	0.0758	0.0408	0.0558	0.0437
	RDRec	T5-Large (0.7B)	0.0067	0.0038	0.0199	0.0130	0.0069	0.0044	0.0132	0.0102
	SLMRec (8→4)	Gemma3 (1.9B)	0.0425	0.0209	0.0757	0.0390	0.0759	0.0399	0.0572	0.0354
	DLLM2Rec	-	0.0292	0.0143	0.0496	0.0242	0.0589	0.0315	0.0224	0.0112
	RLMRec	-	0.0421	0.0263	0.0518	0.0309	0.0239	0.0120	0.0196	0.0122
Ours	R2SLM (SFT)	Gemma3 (1B)	0.0364	0.0191	0.0695	0.0386	0.0777	0.0429	0.0612	0.0468
	R2SLM (Logit KD)	Gemma3 (1B)	0.0276	0.0147	0.0507	0.0273	0.0694	0.0367	0.0536	0.0369
	R2SASRec	-	0.0309	0.0155	0.0526	0.0257	0.0634	0.0336	0.0197	0.0095
	R2END	Text Encoder (0.3B)	**0.0517**	**0.0277**	**0.0958**	**0.0545**	**0.1035**	**0.0587**	**0.0701**	**0.0548**
Improvement			+9.07%	+5.32%	+17.69%	+39.74%	+16.42%	+37.79%	+19.22%	+25.40%

C Additional Experimental Results

Due to space constraints, the primary performance comparison in the main text focuses on top-5 metrics. To provide a more comprehensive evaluation, we present the experimental results for Hit Rate@10 and NDCG@10 in Table 5. Consistent with the results observed in the top-5 metrics, our proposed method outperforms all baseline models across these top-10 metrics as well. These additional findings further validate the robustness of our approach and reaffirm its superiority in various evaluation settings.

D Implementation Details

The text generation process was implemented using vLLM[2], while the embedding extraction was based on SentenceTransformers[3] library. We used the Gemma 3 (Team et al., 2025) model for LLM-based generation and *mxbai-embed-large-v1* (Li and Li, 2024) as the text encoder in the main experiments. All experiments were carried out on a single NVIDIA RTX A6000 GPU with 40GB of VRAM in Ubuntu 22.04.3 LTS environment.

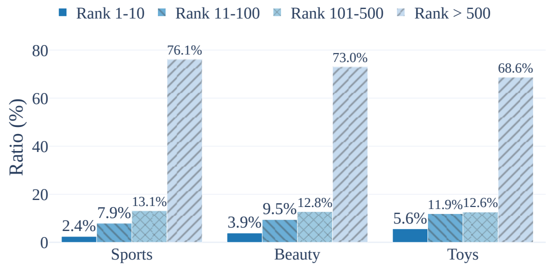
To strictly prevent data leakage, we emphasize that the ground-truth items were utilized as *privileged features* exclusively during the training phase, derived solely from the training split. This follows the standard distillation paradigm where the teacher typically has access to more information than the student. Crucially, during the inference stage, the student model operates without any access to ground-truth labels or future interactions, relying solely on the distilled user representation to retrieve items from the candidate pool.

Our method is trained in two main stages. For the initial reasoning distillation, we fine-tune the student text encoder for a single epoch with a batch size of 16 and a learning rate of $1e-5$. For the subsequent recommendation task training, the projection layer is trained for 10 epochs using a contrastive objective with 99 negative samples per positive instance. We set the temperature for contrastive loss to 0.07. In this stage, we use a batch size of 128, a learning rate of $1e-4$, and an output embedding dimension of 512. The regularization weight λ is set to 0.5. During latency measurement, the experiment was conducted with a batch size of four. A comprehensive list of all hyperparameters and other implementation details is provided in our publicly available repository.

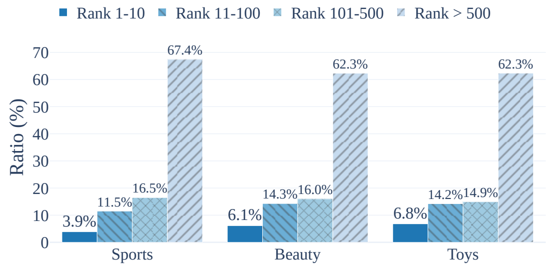
E Performance of Teacher LLM Reasoning

Our research originates from the observation that even with their advanced capabilities, LLMs struggle to identify the correct ground-truth item from a large candidate pool. To substantiate this, Figure 5 presents the performance of applying state-of-the-art foundation models, including *Gemma3-27B-it*[4],

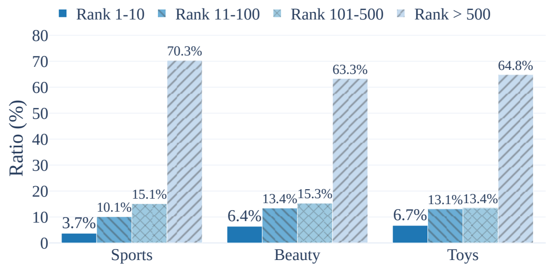
[2]https://docs.vllm.ai/
[3]https://sbert.net/
[4]https://huggingface.co/google/gemma-3-27b-it



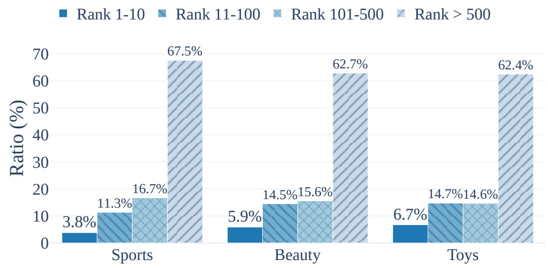
(a) Gemma3-12B



(b) Gemma3-27B



(c) GPT-5.1



(d) Qwen3-Next-80B-Instruct

Figure 5: Misalignment of unguided LLM reasoning. The figures show the recommendation performance of various teacher LLMs (Gemma3, GPT-5.1, and Qwen3-Next) using step-by-step reasoning. Without knowing the ground-truth item, the LLM-generated rationale frequently diverges from actual user behavior.

Table 6: Performance comparison of different distillation methods using the Qwen3-Next-80B-A3B-Instruct as the teacher model. R2END consistently outperforms baselines, demonstrating its generalizability.

Method	Sports				Beauty				Toys				Yelp			
	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10	H@5	N@5	H@10	N@10
SLIM (Teacher)	0.0278	0.0178	0.0429	0.0226	0.0450	0.0296	0.0685	0.0371	0.0538	0.0355	0.0800	0.0439	0.0577	0.0510	0.0657	0.0541
SLIM (Student)	0.0260	0.0167	0.0402	0.0213	0.0482	0.0323	0.0732	0.0403	0.0540	0.0365	0.0809	0.0451	0.0442	0.0368	0.0510	0.0390
R2SLM	0.0215	0.0138	0.0336	0.0177	0.0483	0.0329	0.0730	0.0409	0.0535	0.0361	0.0794	0.0444	0.0422	0.0332	0.0542	0.0370
R2END	**0.0324**	**0.0213**	**0.0487**	**0.0266**	**0.0675**	**0.0472**	**0.0938**	**0.0557**	**0.0754**	**0.0525**	**0.1051**	**0.0621**	**0.0583**	**0.0516**	**0.0680**	**0.0547**

GPT-5.1[5] and *Qwen3-Next-80B-A3B-Instruct*[6], to the SLIM baseline. Despite the massive scale and reasoning power of these models, they surprisingly achieve a top-10 hit rate of less than 7% across all datasets. This highlights a critical misalignment: while LLMs generate semantically plausible reasoning, their general-purpose logic often diverges from the specific, idiosyncratic patterns of actual user behavior in recommendation domains. This empirical evidence provides the fundamental justification for our oracle-guided approach. Since reliance on the teacher's raw predictions leads to the propagation of hallucinations, utilizing the ground truth as a privileged feature is essential to steer the teacher, ensuring that the distilled reasoning is grounded in actual user preferences rather than probable but incorrect guesses.

[5]https://platform.openai.com/docs/models/gpt-5.1

[6]https://huggingface.co/Qwen/Qwen3-Next-80B-A3B-Instruct

F Generalization to Large-Scale Teachers

We further investigated the generalization capability of R2END with respect to significantly larger teacher architectures. We adopted Qwen3-Next-80B-A3B-Instruct as the teacher model. As summarized in Table 6, R2END consistently outperforms baselines, maintaining its effectiveness despite the massive increase in teacher capacity. This demonstrates that our distillation protocol generalizes well across different teacher configurations. Consequently, R2END demonstrates to be a robust solution for transferring the intelligence of foundation models into efficient encoders.

G Verifying Semantic Alignment of Reasoning-Infused Embeddings

We conducted an analysis to verify that our distillation process successfully imbues the student encoder with the semantic essence of the teacher LLM's reasoning. We measured the L2 distance between the student's generated user embeddings and

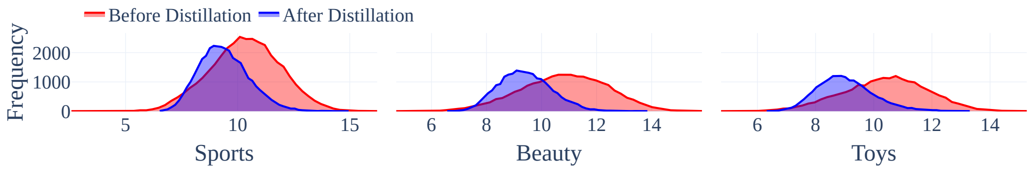


Figure 6: Comparison of L2 distance between user embeddings from encoder and oracle-guided reasoning embeddings in test set, before and after distillation. A smaller distance indicates a higher similarity.

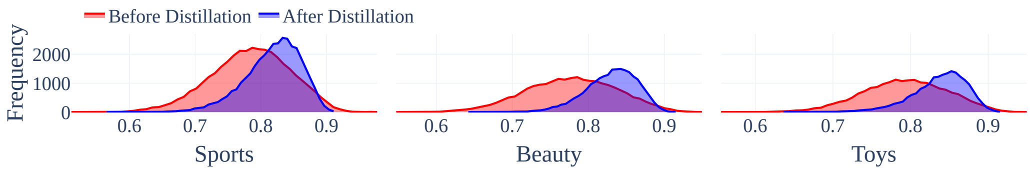


Figure 7: Comparison of cosine similarity between user embeddings from encoder and oracle-guided reasoning embeddings in test set, before and after distillation. A larger value indicates a higher similarity.

the teacher's target reasoning embeddings, both before and after distillation on test set. The resulting distributions, visualized in Figure 6, clearly illustrate the effectiveness of our approach. The blue distribution, representing the reasoning-infused embeddings after distillation, shows a distinct shift towards higher similarity (lower L2 distance) with the teacher's embeddings compared to the pre-distillation state. This result confirms that our student encoder effectively learns to mimic the teacher's aligned reasoning process as intended.

While the improvements in recommendation accuracy are a clear benefit, we also sought to verify that the distillation process was successful in its primary goal: ensuring the student encoder's embeddings capture the semantic essence of the LLM's reasoning. To this end, we analyzed the cosine similarity between the user embeddings generated by the student encoder and the target reasoning embeddings from the teacher, comparing the distributions before and after distillation. The results, illustrated in Figure 7, show a significant increase in cosine similarity across all datasets. This provides strong evidence that our framework not only enhances recommendation accuracy but also successfully mimics the teacher's aligned reasoning as intended.

H Qualitative Analysis: Robustness against Interest Drifts

To demonstrate the robustness of our proposed model in handling dynamic user preferences, we conducted a qualitative case study on users exhibiting significant interest drifts. Table 7 presents representative scenarios where user interests shift abruptly across different categories (e.g., Face → Body) or product types (e.g., Cream → Device). In these challenging cases, the baseline model (SLIM) typically fails to adapt, often overfitting to the dominant category in the user's history. In contrast, our proposed model successfully retrieves the target items, demonstrating superior adaptability.

We attribute this robustness to the integration of aligned reasoning into our training framework. Unlike traditional methods that rely on superficial features, our approach enables the student model to internalize the underlying rationale behind user transitions. By distilling the reasoning capabilities from the teacher model, the student learns to identify latent semantic consistencies, such as shared attributes (e.g., "Organic", "Vitamin C") or functional intents (e.g., "Anti-Aging"), that persist even when the specific product category changes.

For instance, as shown in the table, the model can infer that a preference for organic facial products likely extends to body care (domain transfer), or that a user purchasing anti-aging serums may seek a complementary beauty device (functional intent). This indicates that our method successfully enables the student model to learn complex interest drift patterns inherent in the domain. Consequently, the model becomes resilient to recency bias and capable of flexible reasoning, allowing it to respond effectively to sudden shifts in user intent.

H.1 Impact of Distillation Data Ratio

To evaluate the data efficiency of our proposed framework, we conducted experiments by varying the ratio of training data used for distillation from 20% to 100%. Figure 8 and Figure 9 illustrate the performance trends on the Beauty and Toys datasets, respectively.

The experimental results highlight the robustness

Table 7: Qualitative analysis of user interest drift cases in Beauty dataset. The table compares the user's historical interaction sequence with the ground-truth target item. In all three cases, the proposed model successfully ranked the target item within the Top-10, whereas the SLIM baseline failed to recall it.

Case	User Purchase History (Last 4 Items)	Target Item (Ground Truth)
Case 1 (Face→Body)	1. e.l.f. Eye Primer (**Face**) 2. Thayer Unscented Witch Hazel (**Face**) 3. Wonder Puff Deep Cleansing Puffs (**Face**) 4. Facial Moisturizer - Organic & Natural (**Face**)	**Body** Soap - Organic & Natural
Case 2 (Face→Hair)	1. Vitamin C Serum - Natural & Organic (**Face**) 2. Dead Sea Salt Deep **Hair** Conditioner 3. Adovia Moisturizing Mineral Soap (**Face**) 4. Adovia **Face** Serum - Anti Aging	Argan Oil For **Hair** Treatment & Elixir Serum
Case 3 (Cream→Device)	1. Phytoceramides Anti Aging **Capsules** 2. Shea Butter - 100% Natural **Cream** 3. Integral Beauty's Anti Aging Eye **Cream** 4. Petunia Skincare Revitalize Eye **Serum**	540 Needles Derma Microneedle **Roller** (Device)
Case 4 (Face→Body)	1. Sun*Si'Belle Organic SPF 50 Moisturizer (**Face**) 2. C*Bella Organic Brightening Toner (**Face**) 3. Organic C*Perfect Foaming **Face** Cleanser 4. I*Light Organic Advanced Eye Treatment (**Face**)	Lumi'Essence **Body** Organic Brightening Treatment

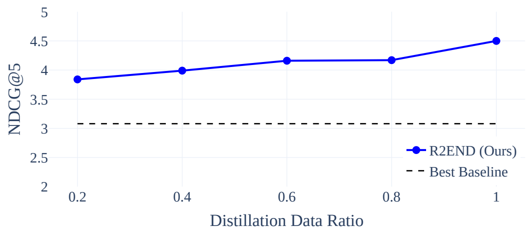


Figure 8: Impact of distillation data ratio on performance (NDCG@5) for the Beauty dataset.

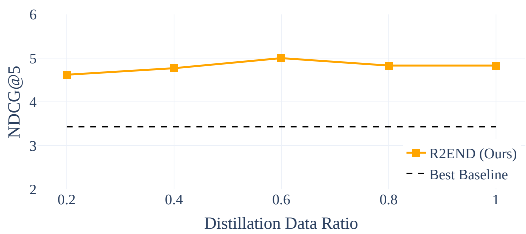


Figure 9: Impact of distillation data ratio on performance (NDCG@5) for the Toys dataset.

of our method even with sparse data. Our proposed method consistently outperforms the best baselines even when utilizing only a subset of the dataset for distillation. As shown in the figures, the model surpasses the baseline performance (represented by the dashed line) even at a training data ratio of just 0.2. These results demonstrate the high learning efficiency of the encoder student, which effectively captures the underlying reasoning patterns from the teacher using limited samples to achieve superior generalization.

Furthermore, these findings suggest a significant reduction in computational costs. Since generating reasoning with a large Teacher LLM is computationally expensive, our approach demonstrates that it is possible to achieve state-of-the-art performance by generating reasoning for only a fraction of the data. This implies that our framework can save substantial computational resources without compromising accuracy.

I Prompt Formats

In this section, we provide a more detailed explanation of the prompt design of oracle-guided reasoning and item descriptions used in our experiments.

I.1 Oracle-guided Reasoning Prompt

Oracle-Guided Reasoning Generation Prompt

```
### Task:
Based on the user's chronological purchase history and the target item, analyze and
summarize user's preferences related to the target item.
Focus on identifying the user's preferences related to the target item from the user's
purchase history.

### User's Purchase History:
{user_purchase_history}

### Target Item:
{target_item}

### Requirements:
- Start with "The user's preferences related to the target item are as follows:"
- Write a single coherent paragraph (max {max_words} words) summarizing the user's
preferences related to the target item.

### Response:
```

Figure 10: Oracle-guided reasoning generation prompt.

To generate the oracle-guided reasoning, we construct a detailed prompt for the teacher LLM. Figure 10 illustrates the prompt format used for the reasoning generation. This prompt provides the model with the user's historical purchase records, which are constructed using up to their 8 most recent interactions from the last 60 days. If a user has no purchases within this period, we use their single most recent transaction. Each interaction in the history text includes the item's purchase date and its associated metadata. Crucially, the prompt also includes the ground-truth next item, which serves as the “oracle.”This setup compels the LLM to generate reasoning that is explicitly aligned with the user’s actual behavior within the recommendation domain. We instruct the LLM to generate this reasoning within a maximum length of 512 words and reduce randomness by setting the temperature to zero. When embedding this reasoning with the text encoder, we incorporate metadata from the ground-truth item to better reflect recency.

I.2 Item Description Prompt

Item Description Generation Prompt

Task:
Analyze the provided metadata and reviews to determine what kinds of users are most likely to prefer the target item. Your response should start with: "Users who prefer [common themes/preferences] would find this item suitable." Incorporate patterns from the reviews, such as favored features, usage scenarios, or functional benefits. Avoid generic statements and ensure your description is grounded in the review content.

Target Item Metadata:
- **Target Item Title**: **{item_title}**
- **Brand**: **{brand}**
- **Original Description**: **{description}**

Reviews of Previously Purchased Items:
{previous_item_reviews}

Requirements:
- Begin with the sentence: "Users who prefer ..."
- Use a single paragraph, no more than **{max_words}** words.
- Focus on inferred user traits, preferences, and realistic use cases.

Response:

Figure 11: Item description generation prompt.

Figure 11 presents the prompt format used for item description generation. Instead of relying on static metadata, our description is generated by leveraging the one-step-prior interactions (at time $T-1$) from the users who purchased that item. This process aims to construct item description based on users behavior, thereby capturing the user preference that leads to a purchase. We set the maximum output length to 512 words and reduce randomness by fixing the temperature at zero. When embedding the final item representation using the text encoder, we concatenate the generated description with the item’s metadata to preserve both semantic and factual aspects. For this item description generation, we used Gemma3-4B, same model used in LLM-based baselines.

J Generated Oracle-Guided Reasoning

We conducted a qualitative analysis to demonstrate that our approach goes beyond capturing surface-level characteristics. By utilizing the reasoning capabilities of LLMs, our framework derives intrinsic motivations and latent preferences from observed behaviors, generating significantly richer semantic information than existing methods. Figure 12 presents examples from four distinct domains (Sports, Beauty, Toys, and Yelp), demonstrating how the model utilizes its extensive world knowledge and reasoning capabilities to interpret user behavior.

First, the analysis highlights the model’s ability to contextualize interactions via world knowledge by abstracting item-level data into higher-level concepts. For instance, in the Sports dataset, the model recognizes that a Bosu Ball and Pull-up Bar are not just isolated items but components of a “well-rounded home gym.”It then uses this domain knowledge to justify the recommendation of Resistance Bands as a complementary tool for “functional fitness,”effectively bridging the gap between distinct equipment types. Furthermore, the model demonstrates a capability to infer latent user intents, even when direct functional similarities are low. In the Beauty dataset, despite the apparent disconnect between purchase history items like tanning lotion or a cleansing system and a target item such as a hair dryer, the model identifies a latent desire for a “polished and enhanced appearance.”This demonstrates the model’s ability to uncover underlying thematic consistency in user behavior across different subcategories.

Moreover, the model excels in nuanced preference identification by detecting specific constraints and context. In the Yelp dataset, beyond simply identifying a preference for “Chinese-inspired cuisine,”it interprets the short time interval between transactions as an indication of “convenience-seeking”behavior or specific “cravings.”Similarly, within the Toys dataset, it distinguishes a specific preference for "construction aspects”over general board games. Collectively, these qualitative results suggest that our approach effectively bridges the semantic gap between user history and target items, providing interpretable and logically sound justifications for recommendations.

Sports

The user's preferences related to the target item are as follows: Based on their recent purchase history, the user demonstrates a clear interest in home-based exercise and fitness equipment, particularly focusing on balance, strength training, and yoga. Their purchases of a Bosu Balance Trainer, yoga blocks, and a doorway pull-up bar all indicate a desire for versatile workout options that can be performed at home without requiring a gym membership. The user seems to favor equipment that allows for a range of exercises, from balance and stability work (Bosu, yoga blocks) to strength training (pull-up bar). While the brands vary, the common thread is a focus on functional fitness and bodyweight exercises. The subsequent purchase of a resistance band set further reinforces this preference for accessible and adaptable exercise tools. Resistance bands complement the existing equipment by offering a portable and scalable way to add intensity and variety to their workouts, allowing for targeted muscle strengthening and rehabilitation. The inclusion of a door anchor and ankle strap suggests an interest in exercises that can be performed in a limited space and engage different muscle groups. Overall, the user appears to be building a well-rounded home gym setup centered around balance, strength, and flexibility, and the resistance band set aligns perfectly with this established fitness profile.

Beauty

The user's preferences related to the target item (John Frieda JF1 Full Volume Hair Dryer) are as follows: While the purchase history doesn't directly involve hair dryers, it reveals a clear interest in beauty and personal care products, specifically focusing on achieving a polished and enhanced appearance. The user previously purchased a self-tanner for achieving a darker skin tone (Millenium Tanning lotion), indicating a desire for a specific aesthetic and a willingness to invest in beauty products to achieve it. Furthermore, the purchase of a Clarisonic cleansing system demonstrates a focus on skincare and maintaining healthy skin, suggesting an appreciation for quality tools and effective routines. This pattern suggests the user is likely seeking beauty tools that contribute to overall appearance and self-care. Given their previous purchases in the beauty category, particularly those focused on skin and appearance enhancement, it's reasonable to infer that the John Frieda hair dryer was purchased to further enhance their beauty routine, likely with an emphasis on volume and styling, aligning with the "Full Volume" designation of the product. The user appears to value products that offer noticeable results and contribute to a desired look, and the hair dryer likely fits within this established preference for beauty-enhancing products.

Toys

The user's preferences related to the target item are as follows: Based on their purchase history from February 4th, 2014, the user demonstrates a clear interest in LEGO building toys. They purchased both the "LEGO City Fire Chief Car 60001" and the "Carcassonne Basic Game" on the same day, indicating a potential interest in both board games and construction toys. However, the LEGO purchase strongly suggests a preference for building sets specifically. Given that the target item, "LEGO Star Wars Corporate Alliance Tank Droid," falls directly within the "Toys & Games, Building Toys, Building Sets" category, it's highly likely the user is interested in LEGO building sets, particularly those with a recognizable theme like Star Wars. While the Carcassonne game shows an interest in games, the simultaneous purchase of a LEGO set suggests building toys hold a stronger appeal. Therefore, the user likely appreciates the creative and construction aspects of LEGO products and would likely be receptive to the target item due to its alignment with their established preference for LEGO building sets.

Yelp

The user's preferences related to the target item are as follows: Based on the provided purchase history, the user demonstrates a clear and consistent preference for Panda Express. They have purchased from Panda Express twice within a very short timeframe (approximately 12 minutes) on November 3rd, 2019. While the specific locations differ (908 E Broadway Ave, Tempe, AZ and 921 N Dobson Rd, Mesa, AZ), the repeated purchases strongly suggest a fondness for this restaurant chain. The categorization of Panda Express as "Chinese" and "Fast Food" indicates the user likely appreciates quick, convenient, and Chinese-inspired cuisine. The user appears to be comfortable with the Panda Express brand, as they have made multiple purchases without showing any preference for other similar restaurants. The close proximity of the purchases in time suggests a potential craving or a desire for a quick meal, further reinforcing their preference for Panda Express as a convenient and satisfying option.

Figure 12: Generated oracle-guided reasoning samples across different domains.