ELICITING INTERACTION-OF-THOUGHT REASONING IN LLM AGENTS FOR PERSONALIZED RECOMMENDATION

Anonymous authors
Paper under double-blind review

ABSTRACT

The emergent reasoning capabilities in Large Language Models (LLMs) have given rise to a surge of research interests in developing personalized agentic systems, particularly in the field of information retrieval. However, it remains a challenge to harness the emergent reasoning capabilities of LLMs for personalized recommendation tasks, largely because general natural language corpora inherently lack recommendation-oriented reasoning traces. To bridge this gap, we introduce R2Rec, a reasoning-enhanced recommendation framework that derives recommendation-oriented reasoning supervision directly from user-item interactions. R2Rec samples interaction chains from the user-item graph to expose high-order collaborative signals and transforms them into structured Interaction-of-Thought traces through a progressive, masked agentic pipeline. By grounding intermediate reasoning in progressively revealed interaction evidence while keeping the final conclusion consistent with the user's actual interaction, these traces provide step-wise supervision for preference reasoning. We then internalize and further refine this capability through post-training: synthesized traces are used for supervised learning of structured reasoning over interaction chains, while recommendation-specific rewards encourage sufficient reasoning decomposition and accurate candidate ranking. Experimental results on three real-world datasets show that R2Rec improves over its backbone LLM by **195.12%** on average across three ranking metrics and outperforms state-of-the-art baselines by **9.87%**. Furthermore, a human study demonstrates the superior interpretability of R2Rec's reasoning, which achieves a **9.07%** improvement over existing methods. Our code is open-source at https://anonymous.4open.science/r/R2Rec-7C5D for reproducibility.

1 INTRODUCTION

The integration of Large Language Models (LLMs) into personalized agentic systems, particularly in the field of information retrieval, has recently garnered substantial research interest Bao et al. (2023); Dai et al. (2023); Geng et al. (2023); Liao et al. (2024); Liu et al. (2023; 2025); Zhao et al. (2024); Zhu et al. (2025). This surge is driven not only by their superior semantic understanding and extensive world knowledge, but also by their powerful complex reasoning capabilities.

Previous LLM-based recommendation approaches typically fall into two paradigms: adapting LLMs as standalone recommenders through prompt tuning Zhang & Wang (2023); Shen et al. (2023); Yang et al. (2021) or fine-tuning Wang et al. (2025a); Bao et al. (2023); Liao et al. (2025); Lin et al. (2024), or leveraging them to enrich traditional models with semantic features derived from textual metadata Wang et al. (2025c); Xi et al. (2026).

However, these methods focus on shallow features, overlooking complex user interests that can only be deciphered through deep reasoning. On the contrary, directly applying Large Reasoning Models like OpenAI o1 Jaech et al. (2024) and DeepSeek-R1 Guo et al. (2025) yields suboptimal results, as their reasoning capabilities are optimized for domains with rich annotated reasoning traces, such as mathematics and coding. In contrast, recommendation systems deal with implicit user behaviors and lack both explicit reasoning traces and natural language supervision, making it challenging to elicit recommendation-oriented reasoning capabilities from LLMs. This creates a critical gap that motivates the need for new approaches capable of eliciting reasoning over user-item interactions.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To address this challenge, we propose R2Rec, a framework that derives recommendation-oriented reasoning supervision directly from user-item interactions. R2Rec begins by sampling interaction chains from the user-item graph to expose high-order collaborative signals in a compact form suitable for LLM reasoning. These chains preserve meaningful relational patterns among users and items, providing structured behavioral evidence beyond isolated interaction histories. We then transform them into structured Interaction-of-Thought traces through a progressive, masked agentic pipeline. By progressively revealing interaction evidence while ensuring that the final conclusion is consistent with the user's actual interaction, the generated traces provide step-wise supervision for preference reasoning. Building on this supervision, we design a recommendation-oriented post-training strategy to internalize and further strengthen such reasoning ability: a set of synthesized traces are used for supervised learning of structured reasoning over interaction chains, while recommendation-specific rewards provide scalable feedback by encouraging sufficient reasoning decomposition and accurate candidate ranking. This combination allows the model to first acquire interaction-aware reasoning patterns from explicit traces and then further refine them through reward-driven optimization without requiring additional reasoning annotations.

We evaluate our method on three real-world datasets Harper & Konstan (2015); Hou et al. (2024a), where R2Rec achieves an average improvement of **9.87%** in HitRatio@1 over SOTA baselines. Notably, our post-training pipeline yields a substantial **195.12%** performance gain over the original backbone LLM. Significantly, this reasoning capability extends beyond interaction chains, enabling robust reasoning from user and item features to maintain high performance even in their absence. Additionally, ablation studies validate the necessity of each training stage and establish interaction chains as the essential reasoning substrate. Furthermore, a human study demonstrates the superior interpretability of R2Rec's reasoning, which achieves a **9.07%** improvement in reasoning quality compared to leading explainable baselines.

In summary, the main contributions of this work include:

- We propose R2Rec, a reasoning-augmented recommendation framework that extracts high-order collaborative signals from user-item graphs through interaction-chain sampling, organizing implicit interaction structures into compact collaborative evidence from which step-wise reasoning supervision can be constructed.
- We develop Interaction-of-Thought supervision through progressive, masked trace generation, grounding intermediate reasoning in interaction evidence while keeping the final conclusion consistent with observed user behavior. We then use these traces for supervised learning and introduce recommendation-specific rewards to further refine reasoning and ranking.
- We conduct experiments on three widely used datasets from different domains of recommendation. Our method consistently outperforms both classical and LLM-based methods, achieving an average performance improvement of 9.87%.

2 Related Works

2.1 Graph-based recommendation

Graph-based methods have demonstrated strong potential in recommendation by modeling user–item interactions as graphs, capturing both direct and high-order relationships Wang et al. (2019); Yang et al. (2018). A key approach leverages Graph Neural Networks (GNNs) such as GraphSAGE Hamilton et al. (2018), NGCF Wang et al. (2019), and LightGCN He et al. (2020) to aggregate neighborhood information for enhanced user and item embeddings. These techniques have also been extended to sequential recommendations, modeling interaction sequences as session or dynamic graphs Zhu et al. (2023); Chang et al. (2023); Zhang et al. (2021); Wang & Cai (2020). While effective, GNN-based methods lack inherent reasoning capabilities and interpretability. Inspired by these approaches, we instead extract interaction chains as structured reasoning paths, enabling LLMs to conduct dynamic, explainable recommendations through sequence-aware reasoning.

2.2 LLM reasoning

Recent advances demonstrate that Large Language Models (LLMs) can significantly enhance reasoning capabilities through reinforcement learning (RL)Yu et al. (2025); Yue et al. (2025); Liu et al.

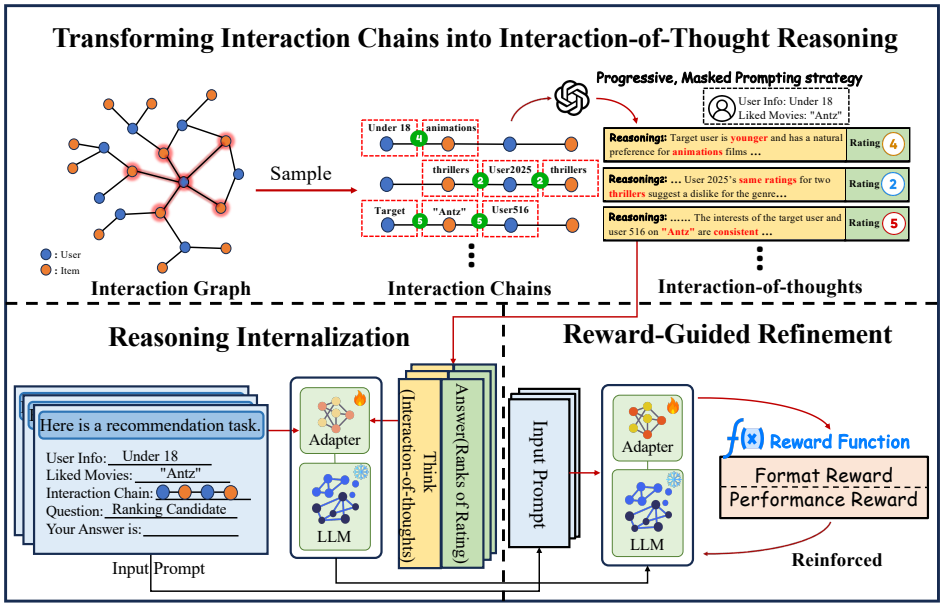


Figure 1: Overview of Reason-to-Recommend(R2Rec) Framework.

(2026); Hu et al. (2025); Wen et al. (2026) and test-time scaling techniques such as Monte Carlo Tree Search (MCTS)Tran et al. (2025); Ha et al. (2025); Liu et al. (2024) and multi-path sampling Wan et al. (2025); Levi (2024); Huang et al. (2026). RL-based post-training methods, exemplified by DeepSeek-R1 Guo et al. (2025) and Kimi k1.5 Team et al., facilitate emergent multi-step reasoning behaviors including planning and self-verification, even without supervised initialization. Hybrid approaches like ReFT integrate supervised fine-tuning with RL to further refine reasoning quality. Despite these advances, LLM reasoning research remains concentrated in structured domains such as mathematics and programming, while recommendation—with implicit user behavior and limited explicit supervision—remains largely underexplored.

2.3 LLM-BASED RECOMMENDATION

Recent advances in Large Language Models (LLMs) have spurred interest in their application to recommendation tasks, with approaches generally falling into two categories. One treats LLMs as standalone recommenders, adapting them through tuning Geng et al. (2023); Chen et al. (2024); Wang et al. (2025a); Liao et al. (2025), prompting Dai et al. (2023), or in-context learning Hou et al. (2024b); Bao et al. (2025). Notable examples include ChatRec Gao et al. (2023), which reformulates user profiles and interactions as prompts; and LLaRA Liao et al. (2024), which treats interaction sequences as additional modalities via curriculum tuning. Some works also use chain-of-thought reasoning, like RecSAVER Tsai et al. (2024), which validates reasoning with quality scores, and ReasoningRec Bismay et al. (2024), which generates explanations and fine-tune smaller models. The other category employs LLMs as enhancers Wang et al. (2025b); Yuan et al. (2023); Xi et al. (2026), leveraging them to generate or extract semantic features like item descriptions to augment traditional recommenders. While effective at integrating external knowledge and semantics, these methods either neglect structured reasoning over user-item interactions or use reasoning limited to single-hop interactions, failing to fully exploit the high-order connectivity Wang et al. (2019) of the user-item interaction graph.

3 METHODS

3.1 OVERVIEW

To enhance the reasoning capabilities of LLMs in recommendation tasks, we propose R2Rec, which derives structured reasoning supervision from user-item interactions. We first construct interaction chains by sampling from the neighborhood of a target user, capturing high-order collaborative signals

for preference reasoning. These chains are then transformed into Interaction-of-Thought traces through a progressive, masked agentic pipeline, providing step-wise supervision grounded in interaction evidence and consistent with observed user behavior. To internalize and refine this capability, we employ post-training with synthesized traces for supervised learning and recommendation-specific rewards for reasoning and ranking optimization. We illustrate the overview of our method in Figure 1.

3.2 Transforming Interaction Chain into Interaction-of-Thought

User-item interaction graphs capture rich structural patterns that reflect users' preferences through their local neighborhoods. However, directly encoding such graphs into LLMs is impractical due to input length limitations and the LLMs' difficulty in processing complex topologies. To address this, we extract interaction chains by sampling from the neighborhood of a target user, forming short sequences that retain meaningful behavioral context while remaining suitable for prompt-based input. To further enhance the interpretability and reasoning potential of these chains, we transform them into step-by-step reasoning chains using a progressive, masked agentic pipeline, enabling LLMs to infer preferences through structured, natural language reasoning. This process is illustrated in Figure 2.

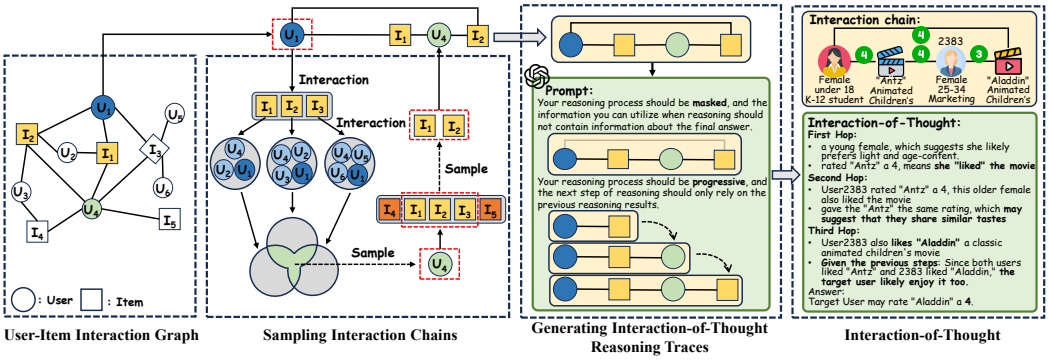


Figure 2: The construction of the interaction chains and the progressive and masked agentic pipeline used to generate the reasoning process.

3.2.1 Interaction Chain

Interaction chains are constructed to capture meaningful local structures within the user–item interaction graph, serving as foundational inputs for downstream reasoning and recommendation by LLMs. However, designing effective chains is non-trivial—naive sampling may produce sequences where the final item is unrelated to the target user, leading to weak preference signals. To address these challenges, we adopt a neighborhood sampling strategy with a closure constraint that ensures the interaction chain forms a valid behavioral loop. Concretely, given a target user u_0, we first collect their interacted item set $\mathcal{I}_{u_0} = i_1, i_2, \ldots, i_n$. For each item i_k in this set, we retrieve the users who have also interacted with it, denoted as $\mathcal{U}_{i_k} = u \mid (u, i_k) \in \mathcal{E}$. To identify candidate users exhibiting similar preferences, we define:

$$\mathcal{U}_{cand} = \bigcup_{\substack{i_p, i_q \in \mathcal{I}_{u_0} \\ i_p \neq i_q}} (\mathcal{U}_{i_p} \cap \mathcal{U}_{i_q}) \setminus \{u_0\} \tag{1}$$

From this candidate set, we randomly sample an intermediate user u_1. Next, we determine the shared item set $\mathcal{I}_{shared} = \mathcal{I}_{u_0} \cap \mathcal{I}_{u_1}$, from which two items i_0, i_1 are selected to form the interaction chain: $u_0 \to i_0 \to u_1 \to i_1 \to u_0$. This construction guarantees that the chain both begins and ends with interactions involving the target user u_0, thereby preserving behavioral coherence and grounding subsequent reasoning in authentic user-item interactions to support accurate preference modeling.

3.2.2 Interaction-of-Thought

To enable LLMs to extract latent patterns from interaction chains, we convert them into structured Interaction-of-Thought using a **progressive**, **masked** agentic pipeline, which encourages the model

to generate step-by-step reasoning by building on prior outputs. Rather than distilling full reasoning traces, which risk invalid outputs and incur costly rejection sampling, we are inspired by the analogy to supervised learning, where optimizers map input-output pairs to parameter updates. Similarly, we treat the LLM as an implicit optimizer that, given a question-answer pair, learns to generate coherent reasoning traces aligned with user interaction behavior.

To realize this, we employ a masked agentic pipeline, wherein the correct answer is injected into the prompt as an outcome-level constraint, guiding the generated trace toward the observed user behavior without being explicitly exposed in the intermediate reasoning. Meanwhile, the progressive prompting design restricts each reasoning step to the interaction evidence revealed up to that point, allowing the LLM to construct the reasoning process step by step. Therefore, we can represent the generation of the reasoning process as:

$$r = \mathcal{LLM}(\mathcal{S}, \mathcal{Q}, \mathcal{A}) \tag{2}$$

where r refers to the reasoning process generated for a specific interaction chain, $\mathcal{Q}$ refers to the query of predicting the start user's rating for the final item in the interaction chain, $\mathcal{A}$ represents the correct rating of the final item by the start user, and $\mathcal{S}$ is the well-designed progressive, masked agentic pipeline. Note that $\mathcal{A}$ acts as an outcome-level constraint on trace synthesis rather than additional interaction evidence; a detailed validity and faithfulness analysis of the generated traces is provided in Appendix A.9.

3.3 LLM POST-TRAINING PIPELINE

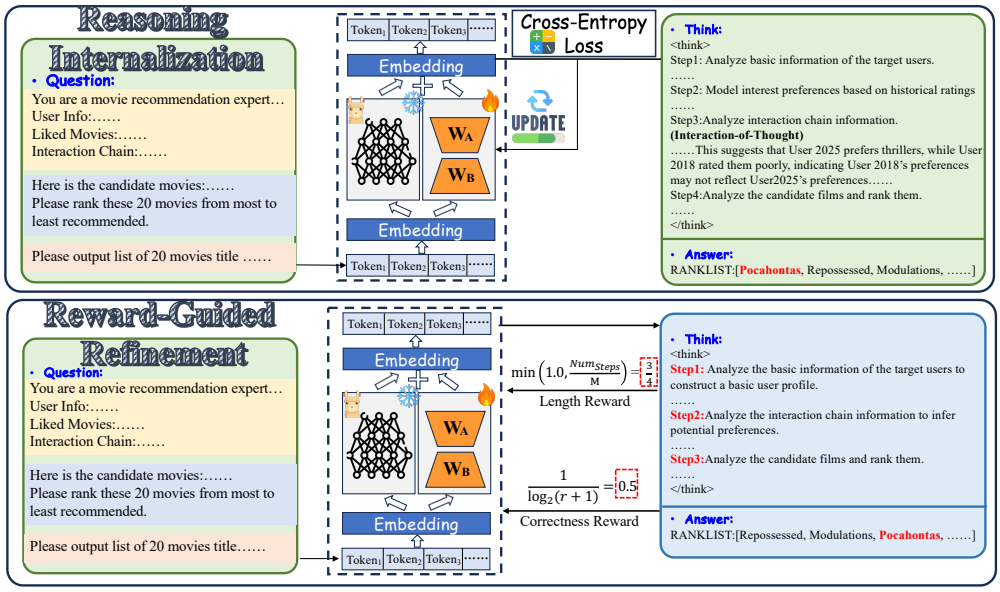


Figure 3: The post-training pipeline for internalizing and refining recommendation-oriented reasoning. W_A and W_B represent two low-rank parameter matrices that are updated during the fine-tuning process.

3.3.1 REASONING TRACE INTERNALIZATION

To equip the LLM with the ability to autonomously reason over interaction chains, we introduce a supervised fine-tuning Hu et al. (2021) stage that serves as a warm-up phase for reasoning-based recommendation, as depicted in Figure 3. Although our progressive, masked prompt design enables the generation of structured reasoning chains, producing such reasoning traces still requires external prompting or strong teacher LLMs, which limits scalability. To address this, we aim to internalize the reasoning process by training the LLM to generate reasoning chains on its own from a limited set of high-quality examples. To construct training data, we adopt the masked prompt strategy described in Section 3.2.2. Specifically, given a recommendation query Q, which asks the LLM to predict a user's preference for an item based on an interaction chain, and a correct answer A, we prompt the LLM

to generate an intermediate reasoning chain R that logically connects Q and A. Importantly, while A is embedded in the prompt, the LLM is instructed to ignore it explicitly and instead simulate the reasoning process as if it were solving the problem from scratch. This allows the generated reasoning to be guided yet faithful to the underlying interaction context. Each training instance is structured as a triplet (Q, R, A), which is then used to optimize the LLM's generation behavior through supervised learning. The training objective is formalized as:

$$\theta^* = \arg\min_{\theta} \mathbb{E}_{(Q,R,A)\sim\mathcal{D}} \mathcal{L}(\text{LLM}_\theta(Q), R, A) \tag{3}$$

Here, $\mathcal{D}$ denotes the constructed training dataset and $\mathcal{L}$ is the loss function encouraging the LLM to produce reasoning chains and answers aligned with the gold supervision.

3.3.2 REWARD-GUIDED REASONING REFINEMENT

To further enhance the LLM's ability to reason over interaction chains and improve its generalization beyond limited annotated examples, we introduce a reinforcement learning (RL) stage. This stage is designed to provide scalable supervision signals by allowing the LLM to refine its reasoning strategies through interaction with a reward function, without relying on additional labeled reasoning chains. In this way, RL complements the supervised fine-tuning phase by enabling continuous self-improvement through feedback. We adopt the Group Relative Policy Optimization (GRPO) algorithm Shao et al. (2024), which stabilizes policy updates by computing group-normalized advantages over multiple sampled completions without requiring a separate value network. The GRPO objective is defined as:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim P(Q),\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)} \tag{4}$$
$$\left[\frac{1}{G}\sum_{i=1}^{G}\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}A_i,\ \ \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}, 1-\epsilon, 1+\epsilon\right)A_i\right) - \beta D_{\text{KL}}(\pi_\theta \parallel \pi_{\text{ref}})\right]$$

where A_i denotes the group-relative advantage, computed by normalizing the rewards of actions within the same group, thereby stabilizing training and improving efficiency. To effectively guide the optimization, we define the total reward used in RL training as the combination of two components: the reasoning length reward and the ranking correctness reward.

Reasoning Length Reward. This reward encourages the model to decompose its decision process into a sufficient number of reasoning steps. Since the input prompt contains $M = 4$ major evidence blocks (user profile, recent preference history, interaction chains, and candidate items), we use M as a coarse structural prior:

$$\text{Reward}_{\text{length}} = \min\left(1.0, \frac{Num_{\text{steps}}}{M}\right), \quad M = 4 \tag{5}$$

The reward saturates after M steps, so additional verbosity is not positively incentivized.

Ranking Correctness Reward. This reward reflects how well the ground-truth item is ranked in the output list. Let r denote the rank of the ground-truth item in the generated ranking list:

$$\text{Reward}_{\text{correctness}} = \begin{cases} \dfrac{1}{\log_2(r+1)}, & \text{if the ground-truth item is ranked at position } r, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

Final Reward. Since both components are normalized to $[0, 1]$, we combine them with equal weights:

$$\text{Reward}_{\text{total}} = \text{Reward}_{\text{length}} + \text{Reward}_{\text{correctness}} \tag{7}$$

A component-wise ablation of the two rewards is reported in Appendix A.13.2.

4 EXPERIMENTS

In this section, we evaluate R2Rec across multiple dimensions, including performance comparisons against baselines, ablation studies on framework components, cross-domain transferability, and case studies.

4.1 Experimental Settings

4.1.1 Datasets

We conduct experiments on three real-world datasets: MovieLens-1M Harper & Konstan (2015), Amazon Book, and Amazon Electronics Hou et al. (2024a). For each dataset, we retain users with at least six ratings above 3 and sort their ratings chronologically. The most recent positively rated item is used as the ground-truth. Following previous studies Liao et al. (2024); Chen et al. (2024); Kong et al. (2024), we create a candidate set by combining the positive item with 19 randomly sampled negative items and evaluate the LLM's ranking. Each dataset includes 1,000 randomly sampled users for evaluation. More details about the datasets are provided in the Appendix A.2. Additional enlarged-candidate and full-catalog retrieval-and-reranking evaluations are provided in Appendix A.8.

4.1.2 Implementation Details

We use DeepSeek-R1-Distill-Llama-8B as the backbone LLM. During fine-tuning, we apply Low-Rank Adaptation (LoRA) with the AdamW optimizer in both the SFT and RL stages. More details can be found in Appendix A.3.

4.2 Performance Comparsion

In this section, we evaluate R2Rec against classical and LLM-based baselines on MovieLens, Amazon Book, and Amazon Electronics using HitRatio@1, HitRatio@3, and NDCG@3. (We omit NDCG@1 as it is equivalent to HitRatio@1 due to the single positive item in each candidate set.)

4.2.1 Baselines

- Classic Methods: **GraphSage** Hamilton et al. (2018), **NGCF** Wang et al. (2019), **LightGCN** He et al. (2020) in graph neural networks for recommendation, as well as **GRU4Rec** Hidasi et al. (2016), **Caser** Tang & Wang (2018) and **SASRec** Kang & McAuley (2018), which are RNN-based, CNN-based and attention-based recommenders, respectively.
- LM-based Methods: We thoroughly compare R2Rec with four categories of LLM-based methods: vanilla LLMs(**DeepSeek-R1-Distill-Llama-8B**, **GPT-4o-mini**, and **o4-mini**), LLM4Rec approaches(**ChatRec** Gao et al. (2023), **TALLRec** Bao et al. (2023), **LLaRA** Liao et al. (2024), and **S-DPO** Chen et al. (2024)), reasoning-enhanced methods(**Rec-SAVER** Tsai et al. (2024), **Reason4Rec** Fang et al. (2025), and **RecZero** Kong et al. (2025)) and graph-reasoning method (**GraphAdapter** Huang et al. (2024)). See detailed introduction of LLM-based methods in Appendix A.4.

4.2.2 Results

We present the experimental results in Table 1, where each result is the average of three independent runs. The observation can be summarized as follows.

Classical recommenders consistently underperform R2Rec, as they mainly model observable interaction patterns without explicit semantic reasoning. LLM-based methods are more competitive by leveraging world knowledge, semantic understanding, and task adaptation; notably, o4-mini outperforms GPT-4o-mini, suggesting the benefit of stronger reasoning. LLM4Rec and reasoning-enhanced methods further gain from recommendation alignment and chain-of-thought reasoning, while GraphAdapter benefits from multi-hop interaction signals. Overall, these results highlight the complementary value of semantics, reasoning, and structural interaction information for recommendation.

R2Rec outperforms all baseline models across the three datasets, achieving an average improvement of 9.87% in HitRatio@1 compared to the second-best baseline. Specifically, R2Rec attains the highest HitRatio@1 scores of 0.411, 0.591, and 0.467 on MovieLens, Amazon-Book, and Amazon-Electronics, respectively, with relative improvements of 6.75%, 9.04%, and 13.90% over the second-best baselines on each dataset. These results demonstrate that utilizing the reasoning ability of LLMs to mine the rich signals embedded in interaction chains can effectively enhance the recommendation performance of LLMs. Further detailed analyses are provided in the Appendix, including statistical

Table 1: The Results of R2Rec compared with classical recommendation models, Language Model-based Recommendation and Reasoning enhanced Recommendation. Bold and underline indicate the best and second-best performance. * denotes statistical significance (p-value < 0.01) compared to the secondbest result. HR denotes HitRatio.

	Method	MovieLens			Amazon Book			Amazon Electronics		
		HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
Classical	GraphSage	0.313	0.491	0.411	0.321	0.502	0.465	0.318	0.516	0.443
	NGCF	0.358	<u>0.512</u>	<u>0.472</u>	0.455	0.521	0.497	0.372	0.524	0.451
	LightGCN	0.276	0.491	0.435	0.362	0.537	0.465	0.379	<u>0.563</u>	0.487
	GRU4Rec	0.376	0.504	0.452	0.495	0.572	0.527	0.386	0.521	<u>0.498</u>
	Caser	0.319	0.496	0.421	0.422	<u>0.631</u>	0.544	0.385	0.518	0.496
	SASRec	0.380	0.510	0.452	0.531	0.592	0.563	0.391	0.514	0.453
LM-based	DeepSeek-R1-8B	0.156	0.299	0.237	0.160	0.324	0.255	0.072	0.156	0.118
	GPT-4o-mini	0.240	0.374	0.317	0.344	0.502	0.436	0.222	0.318	0.276
	o4-mini	0.274	0.419	0.357	0.466	0.596	0.541	0.282	0.372	0.335
	ChatRec	0.230	0.340	0.293	0.304	0.408	0.362	0.258	0.370	0.315
	TallRec	0.318	0.442	0.401	0.412	0.478	0.421	0.342	0.432	0.368
	LLaRA	0.371	0.473	0.454	0.528	0.583	<u>0.556</u>	0.403	0.526	0.491
	S-DPO	<u>0.385</u>	0.502	0.470	<u>0.542</u>	0.616	0.549	<u>0.410</u>	0.549	0.497
	Graphadapter	0.353	0.473	0.455	0.483	0.537	0.498	0.366	0.494	0.468
Reasoning Enhanced	Rec-SAVER	0.335	0.462	0.427	0.476	0.573	0.524	0.347	0.511	0.446
	Reason4Rec	-	-	-	0.514	0.603	0.527	0.396	0.527	0.489
	RecZero	0.364	0.472	0.447	0.524	0.610	0.531	0.404	0.535	0.486
	R2Rec	**0.411***	**0.549***	**0.493***	**0.591***	**0.664***	**0.632***	**0.467***	**0.602***	**0.545***

significance (A.5), model scalability (A.6), sparse interactions (A.7), enlarged and full-catalog candidate spaces (A.8), validity and faithfulness of reasoning traces (A.9), and backbone LLM generality (A.10).

4.3 Ablation Study

In order to verify the validity of our designs, we conduct ablation studies regarding both interaction chain features in prompt and the stages of training pipeline. We show the results in Table 2.

Table 2: Ablation Study on Prompt Features and Training Stages. HR denotes HitRatio.

	Method	MovieLens			Amazon Book			Amazon Electronics		
		HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
Prompt	Full	0.230	0.381	0.317	0.366	0.494	0.440	0.192	0.326	0.268
	w/o Reasoning Chain	0.186	0.327	0.267	0.278	0.420	0.361	0.172	0.256	0.219
	w/o Interaction Chain	0.156	0.299	0.237	0.160	0.324	0.255	0.072	0.156	0.118
Training Stage	Full	0.411	0.549	0.493	0.591	0.664	0.632	0.467	0.602	0.545
	w/o RL	0.320	0.453	0.397	0.528	0.608	0.574	0.436	0.490	0.467
	w/o SFT	0.162	0.277	0.225	0.254	0.343	0.300	0.116	0.164	0.142

4.3.1 Ablation Study of Interaction Chain

The upper block of Table 2 investigates the impact of removing components from the prompt. We observe that eliminating the reasoning chain leads to consistent performance drops across all datasets, indicating that explicitly modeling reasoning is beneficial. More notably, removing the interaction chain results in the most substantial degradation—e.g., on Amazon Electronics, HitRatio@1 drops from 0.192 to 0.072—suggesting that the interaction chain serves as the essential substrate for reasoning. In other words, the interaction chain provides the behavioral context from which reasoning chains can be derived. Without this foundation, the LLM lacks the necessary signals to support preference inference. A finer-grained analysis of what makes a good interaction chain, together with the impact of chain number and length, is provided in Appendix A.11 and Appendix A.12.

4.3.2 Ablation Study of Training Stage

The bottom part of Table 2 shows the impact of removing training stages: omitting supervised fine-tuning (SFT) causes a significant drop in performance (e.g., HitRatio@1 on MovieLens falls from 0.411 to 0.162), highlighting SFT's critical role in warm-starting the model's reasoning ability using limited curated data. Removing reinforcement learning (RL) also degrades results, though

less severely, demonstrating RL's value in providing scalable supervision by enabling the LLM to refine reasoning via reward feedback without labeled data. Further, Figure 6 compares reward curves during RL training with and without SFT, showing that the SFT-initialized model achieves higher and more stable rewards, while training from scratch leads to instability and suboptimal convergence. Further analyses of RL training dynamics and reward-component contributions are provided in Appendix A.13.1 and Appendix A.13.2, respectively.

4.4 Transfer Evaluation

Table 3: Cross-Domain Transferability of R2Rec. HR denotes HitRatio.

		Domain of Test Data								
		MovieLens			Amazon Book			Amazon Electronics		
		HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
Domain of Training Data	MovieLens	0.411	0.549	0.493	0.508	0.619	0.541	0.300	0.364	0.329
	Amazon Book	0.302	0.446	0.386	0.591	0.664	0.632	0.434	0.481	0.460
	Amazon Electronics	0.239	0.321	0.290	0.496	0.597	0.533	0.467	0.602	0.545

To evaluate the transferability of our method, we perform cross-domain experiments by training on one dataset and testing on another. The results are shown in Table 3. For example, the LLM trained on MovieLens achieves an HitRatio@1 of 0.508 when tested on Amazon Book, which is competitive and even outperforms several in-domain baselines. This indicates that the LLM acquires a transferable reasoning ability over interaction chains, rather than overfitting to domain-specific user preferences. Such capability allows it to generalize across different domains by leveraging structural patterns in user-item interactions rather than relying solely on memorized user behavior. A comparison against classical baselines under the same transfer protocol is provided in Appendix A.14.

4.5 User Study

To evaluate the quality of the reasoning processes generated by different recommender systems, we conducted a user study. We compared the reasoning processes of our method, R2Rec, with those of two explainable recommendation models: Reason4Rec and Rec-SAVER. Human evaluators were asked to assess the reasoning processes on three dimensions: Relevance, Interpretability, and Overall Quality. Detailed settings of the user study can be found in Appendix A.15.

Table 4: Human Evaluation Result

Method	Relevance	Interpretability	Overall Quality
Rec-SAVER	3.647	3.235	3.529
Reason4Rec	3.882	3.861	3.864
RecZero	3.956	3.947	4.031
R2Rec	4.357	4.265	4.394

The results of this user study are presented in Table 4. From the table, we observe that the reasoning process generated by R2Rec consistently received the highest scores across all three dimensions. Specifically, human evaluators noted that the "explicit reasoning traces over interactions" allow for an "in-depth evaluation" of user intent, thereby providing superior interpretability. Furthermore, the model's capacity to leverage "collaborative connections" between users ensures the generation of "highly relevant" recommendations. To intuitively demonstrate the quality and interpretability of our reasoning process, we provide a detailed case study in Appendix A.15.

5 Conclusions

In this paper, we address the challenge of the lack of explicit reasoning traces in recommendation systems. To tackle this, we propose R2Rec, a reasoning-enhanced recommendation framework that leverages interaction chains to serve as the foundation for structured reasoning. By transforming these chains into structured Interaction-of-Thought sequences via a progressive, masked prompting strategy, we enable LLMs to perform stepwise reasoning grounded in user-item interaction context. Through extensive experiments, we demonstrate that our framework significantly outperforms state-of-the-art baselines while enhancing interpretability. Future work will focus on scaling this approach through retrieval-augmented prompting and efficient encoding strategies.

AI USE STATEMENT

Generative AI tools were used to aid and polish the writing, improve presentation structure, and identify potentially relevant literature; all cited works were verified by the authors. The research methodology, experiments, analyses, and final conclusions were conducted and validated by the authors. We take responsibility for the final content of this work.

REFERENCES

Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, RecSys '23, pp. 1007–1014. ACM, September 2023. doi: 10.1145/3604915.3608857. URL http://dx.doi.org/10.1145/3604915.3608857.

Keqin Bao, Ming Yan, Yang Zhang, Jizhi Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Customizing in-context learning for dynamic interest adaption in llm-based recommendation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 14278–14291, 2025.

Millennium Bismay, Xiangjue Dong, and James Caverlee. Reasoningrec: Bridging personalized recommendations and human-interpretable explanations through llm reasoning. *arXiv preprint arXiv:2410.23180*, 2024.

Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. Sequential recommendation with graph neural networks, 2023. URL https://arxiv.org/abs/2106.14226.

Yuxin Chen, Junfei Tan, An Zhang, Zhengyi Yang, Leheng Sheng, Enzhi Zhang, Xiang Wang, and Tat-Seng Chua. On softmax direct preference optimization for recommendation. *Advances in Neural Information Processing Systems*, 37:27463–27489, 2024.

Sunhao Dai, Ninglu Shao, Haiyuan Zhao, Weijie Yu, Zihua Si, Chen Xu, Zhongxiang Sun, Xiao Zhang, and Jun Xu. Uncovering chatgpt's capabilities in recommender systems. In *Proceedings of the 17th ACM Conference on Recommender Systems*, RecSys '23, pp. 1126–1132. ACM, September 2023. doi: 10.1145/3604915.3610646. URL http://dx.doi.org/10.1145/3604915.3610646.

Yi Fang, Wenjie Wang, Yang Zhang, Fengbin Zhu, Qifan Wang, Fuli Feng, and Xiangnan He. Reason4rec: Large language models for recommendation with deliberative user preference alignment. *arXiv preprint arXiv:2502.02061*, 2025.

Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. Chat-rec: Towards interactive and explainable llms-augmented recommender system, 2023. URL https://arxiv.org/abs/2303.14524.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5), 2023. URL https://arxiv.org/abs/2203.13366.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Rui Ha, Chaozhuo Li, Rui Pu, Litian Zhang, Xi Zhang, and Sen Su. Dsg-mcts: A dynamic strategy-guided monte carlo tree search for diversified reasoning in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 10541–10555, 2025.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs, 2018. URL https://arxiv.org/abs/1706.02216.

F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4), December 2015. ISSN 2160-6455. doi: 10.1145/2827872. URL https://doi.org/10.1145/2827872.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation, 2020. URL https://arxiv.org/abs/2002.02126.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks, 2016. URL https://arxiv.org/abs/1511.06939.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*, 2024a.

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

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Jian Hu, Jason Klein Liu, and Wei Shen. Reinforce++: An efficient rlhf algorithm with robustness to both prompt and reward models, 2025. URL https://arxiv.org/abs/2501.03262.

Baihe Huang, Shanda Li, Tianhao Wu, Yiming Yang, Ameet Talwalkar, Kannan Ramchandran, Michael Jordan, and Jiantao Jiao. Sample complexity and representation ability of test-time scaling paradigms. In *International Conference on Learning Representations*, volume 2026, pp. 54262–54298, 2026.

Xuanwen Huang, Kaiqiao Han, Yang Yang, Dezheng Bao, Quanjin Tao, Ziwei Chai, and Qi Zhu. Can gnn be good adapter for llms? In *Proceedings of the ACM Web Conference 2024*, pp. 893–904, 2024.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation, 2018. URL https://arxiv.org/abs/1808.09781.

Xiaoyu Kong, Jiancan Wu, An Zhang, Leheng Sheng, Hui Lin, Xiang Wang, and Xiangnan He. Customizing language models with instance-wise lora for sequential recommendation. *Advances in Neural Information Processing Systems*, 37:113072–113095, 2024.

Xiaoyu Kong, Junguang Jiang, Bin Liu, Ziru Xu, Han Zhu, Jian Xu, Bo Zheng, Jiancan Wu, and Xiang Wang. Think before recommendation: Autonomous reasoning-enhanced recommender. *arXiv preprint arXiv:2510.23077*, 2025.

Noam Levi. A simple model of inference scaling laws, 2024. URL https://arxiv.org/abs/2410.16377.

Jiayi Liao, Sihang Li, Zhengyi Yang, Jiancan Wu, Yancheng Yuan, Xiang Wang, and Xiangnan He. Llara: Large language-recommendation assistant, 2024. URL https://arxiv.org/abs/2312.02445.

Jiayi Liao, Ruobing Xie, Sihang Li, Xiang Wang, Xingwu Sun, Zhanhui Kang, and Xiangnan He. Multi-grained patch training for efficient llm-based recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1572–1581, 2025.

Xinyu Lin, Wenjie Wang, Yongqi Li, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. Bridging items and language: A transition paradigm for large language model-based recommendation, 2024. URL https://arxiv.org/abs/2310.06491.

Jiacheng Liu, Andrew Cohen, Ramakanth Pasunuru, Yejin Choi, Hannaneh Hajishirzi, and Asli Celikyilmaz. Don't throw away your value model! generating more preferable text with value-guided monte-carlo tree search decoding, 2024. URL https://arxiv.org/abs/2309.15028.

Jiahao Liu, Xueshuo Yan, Dongsheng Li, Guangping Zhang, Hansu Gu, Peng Zhang, Tun Lu, Li Shang, and Ning Gu. Improving llm-powered recommendations with personalized information, 2025. URL https://arxiv.org/abs/2502.13845.

Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. Is chatgpt a good recommender? a preliminary study, 2023. URL https://arxiv.org/abs/2304.10149.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. Prorl: Prolonged reinforcement learning expands reasoning boundaries in large language models. *Advances in Neural Information Processing Systems*, 38:17998–18031, 2026.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Tianshu Shen, Jiaru Li, Mohamed Reda Bouadjenek, Zheda Mai, and Scott Sanner. Towards understanding and mitigating unintended biases in language model-driven conversational recommendation. *Information Processing & Management*, 60(1):103139, 2023. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2022.103139. URL https://www.sciencedirect.com/science/article/pii/S0306457322002400.

Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding, 2018. URL https://arxiv.org/abs/1809.07426.

Kimi Team, A Du, B Gao, B Xing, C Jiang, C Chen, C Li, C Xiao, C Du, C Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms, 2025. *URL https://arxiv. org/abs/2501.12599*.

Hieu Tran, Zonghai Yao, Zhichao Yang, Junda Wang, Yifan Zhang, Shuo Han, Feiyun Ouyang, and Hong Yu. Rare: Retrieval-augmented reasoning enhancement for large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 18305–18330, 2025.

Alicia Y Tsai, Adam Kraft, Long Jin, Chenwei Cai, Anahita Hosseini, Taibai Xu, Zemin Zhang, Lichan Hong, Ed H Chi, and Xinyang Yi. Leveraging llm reasoning enhances personalized recommender systems. *arXiv preprint arXiv:2408.00802*, 2024.

Guangya Wan, Yuqi Wu, Jie Chen, and Sheng Li. Reasoning aware self-consistency: Leveraging reasoning paths for efficient llm sampling. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3613–3635, 2025.

Baocheng Wang and Wentao Cai. Knowledge-enhanced graph neural networks for sequential recommendation. *Information*, 11(8), 2020. ISSN 2078-2489. doi: 10.3390/info11080388. URL https://www.mdpi.com/2078-2489/11/8/388.

Bohao Wang, Feng Liu, Jiawei Chen, Xingyu Lou, Changwang Zhang, Jun Wang, Yuegang Sun, Yan Feng, Chun Chen, and Can Wang. Msl: Not all tokens are what you need for tuning llm as a recommender. In *Proceedings of the 48th international ACM SIGIR conference on research and development in information retrieval*, pp. 1912–1922, 2025a.

Ke Wang, Ji Zhang, and Kuan Liu. Enhancing cross-domain recommendation with plug-in contrastive representations from large language models. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1799–1809, 2025b.

Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '19, pp. 165–174. ACM, July 2019. doi: 10.1145/3331184.3331267. URL http://dx.doi.org/10.1145/3331184.3331267.

Yuhao Wang, Junwei Pan, Pengyue Jia, Wanyu Wang, Maolin Wang, Zhixiang Feng, Xiaotian Li, Jie Jiang, and Xiangyu Zhao. Pre-train, align, and disentangle: Empowering sequential recommendation with large language models. In *Proceedings of the 48th international ACM SIGIR conference on research and development in information retrieval*, pp. 1455–1465, 2025c.

Xumeng Wen, Zihan Liu, Shun Zheng, Shengyu Ye, Zhirong Wu, Yang Wang, Zhijian Xu, Xiao Liang, Junjie Li, Ziming Miao, et al. Reinforcement learning with verifiable rewards implicitly incentivizes correct reasoning in base llms. In *International Conference on Learning Representations*, volume 2026, pp. 49450–49483, 2026.

Yunjia Xi, Menghui Zhu, Jianghao Lin, Bo Chen, Ruiming Tang, Yong Yu, and Weinan Zhang. Modular representation compression: Adapting llm representations for efficient and effective recommendation. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2027–2038, 2026.

Bowen Yang, Cong Han, Yu Li, Lei Zuo, and Zhou Yu. Improving conversational recommendation systems' quality with context-aware item meta information, 2021. URL https://arxiv.org/abs/2112.08140.

Jheng-Hong Yang, Chih-Ming Chen, Chuan-Ju Wang, and Ming-Feng Tsai. Hop-rec: high-order proximity for implicit recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, pp. 140–144, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450359016. doi: 10.1145/3240323.3240381. URL https://doi.org/10.1145/3240323.3240381.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Weinan Dai, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025. URL https://arxiv.org/abs/2503.14476.

Zheng Yuan, Fajie Yuan, Yu Song, Youhua Li, Junchen Fu, Fei Yang, Yunzhu Pan, and Yongxin Ni. Where to go next for recommender systems? id- vs. modality-based recommender models revisited, 2023. URL https://arxiv.org/abs/2303.13835.

Yu Yue, Yufeng Yuan, Qiying Yu, Xiaochen Zuo, Ruofei Zhu, Wenyuan Xu, Jiaze Chen, Chengyi Wang, TianTian Fan, Zhengyin Du, Xiangpeng Wei, Xiangyu Yu, Gaohong Liu, Juncai Liu, Lingjun Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Ru Zhang, Xin Liu, Mingxuan Wang, Yonghui Wu, and Lin Yan. Vapo: Efficient and reliable reinforcement learning for advanced reasoning tasks, 2025. URL https://arxiv.org/abs/2504.05118.

Mengqi Zhang, Shu Wu, Xueli Yu, Qiang Liu, and Liang Wang. Dynamic graph neural networks for sequential recommendation, 2021. URL https://arxiv.org/abs/2104.07368.

Zizhuo Zhang and Bang Wang. Prompt learning for news recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, pp. 227–237. ACM, July 2023. doi: 10.1145/3539618.3591752. URL http://dx.doi.org/10.1145/3539618.3591752.

Yuyue Zhao, Jiancan Wu, Xiang Wang, Wei Tang, Dingxian Wang, and Maarten de Rijke. Let me do it for you: Towards llm empowered recommendation via tool learning. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2024, pp. 1796–1806. ACM, July 2024. doi: 10.1145/3626772.3657828. URL http://dx.doi.org/10.1145/3626772.3657828.

Tianyu Zhu, Leilei Sun, and Guoqing Chen. Graph-based embedding smoothing for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):496–508, 2023. doi: 10.1109/TKDE.2021.3073411.

Yaochen Zhu, Chao Wan, Harald Steck, Dawen Liang, Yesu Feng, Nathan Kallus, and Jundong Li. Collaborative retrieval for large language model-based conversational recommender systems, 2025. URL https://arxiv.org/abs/2502.14137.

A APPENDIX

A.1 THE PROGRESSIVE, MASKED PROMPTING STRATEGY TO GENERATE INTERACTION-OF-THOUGHT

The specific prompt to generate Interaction-of-Thought is as follows(Using the Movielens dataset as an example).

Prompt

You are a movie recommendation expert. I will give you a user-movie interaction chain, a question and a correct answer. The interaction chain contains the user's rating of a certain movie and the basic information of the user/movie. Movie ratings are integers from 1 to 5, with 5 standing for very liked, 4 standing for liked, 3 standing for neutral, 2 standing for disliked and 1 standing for very disliked.
The following is the form of the interaction chain that shows the common viewing behavior and ratings among users, and the basic information of the user/movie is also recorded in this interaction chain:
(Target user)(Basic information of Target user) – (Rating 5) – (Movie A)(Basic information of Movie A) – (Rating 4) – (User 1234)(Basic information of User1234) – (Rating 2) – (Movie B)(Basic information of Movie B) – (Rating 3) – (Target user)
The meaning of the above interaction chain is:
The target user has watched the Movie A and gave it a score of 5, while user1234 has also watched Movie A and gave it a score of 4, and user1234 has also watched the Movie B and gave it a score of 2. This target user has watched the Movie B as well and gives it a score of 3.
Your task is to pretend to reason step by step about this interaction chain without knowing the given answer in advance, independently analyze the information of each jump in the interaction chain, and gradually obtain the final answer. The reasoning process must comply with the following requirements:
1. Your reasoning process should be progressive, and the next step of reasoning should only rely on the previous reasoning results.
2. Your reasoning process should be masked. When reasoning about the information of a certain hop in the interaction chain, the only information you can utilize is the information before that hop, and you cannot utilize the information after that hop.
3. Your reasoning process should not contain any information about the final answer. The answers provided are only to guide your reasoning process.
4. The answer obtained based on your reasoning process should be consistent with the correct one.
Next, I will provide you with a user-movie interaction chain, a question and a correct answer:
Interaction chain:
{Interaction Chain}
Question:
What is the rating of the movie *{Final movie of the interaction chain}* given by the target users?
Answer:
Rating *{True rating of the target user given to the final movie}*
Please output your reasoning process and the final answer obtained based on the reasoning process:
Reasoning process:
Final answer:

A.2 MORE DETAILS OF THE DATASETS

- **MovieLens-1M** Harper & Konstan (2015) is a commonly-used movie recommendation dataset contains user demographics, movie attributes and user-movie ratings.
- **Amazon** Hou et al. (2024a) is derived from Amazon, a leading global e-commerce platform, which provides rich information including user reviews, item metadata, and user-item links. In our work, we select datasets from two domains: Book and Electronics.

Table 5 presents the detailed statistics of the datasets.

Table 5: Statistics of Datasets.

Dataset	Movielens	Book	Electronics
# User	6,040	776,370	1,641,026
# Item	3,900	495,063	368,228
# Interaction	1,000,209	9,488,297	15,473,536

A.3 More details of implementation

We use DeepSeek-R1-Distill-Llama-8B Guo et al. (2025) as the backbone LLM, with temperature set to 0.6 and top-p to 0.9. For each user, the prompt includes demographic features (if available), five recently liked items, five sampled interaction chains, and a 20-item candidate set. During the SFT stage, we apply Low-Rank Adaptation (LoRA) Hu et al. (2021) to fine-tune the LLM using the AdamW optimizer with a learning rate of 5e-5, for 10 epochs and 500 training samples. In the RL stage, we further train the LoRA adapter obtained from SFT with the same optimizer, using a smaller learning rate of 5e-6, for 1 epoch and 500 training samples.

A.4 More information about LLM-based baselines

- **DeepSeek-R1-Distill-Llama-8B** Guo et al. (2025) is the backbone model of R2Rec.
- **GPT-4o-mini** and **o4-mini** are closed-source LLMs released by OpenAI.
- **ChatRec** Gao et al. (2023) reformulates user data into prompts for recommendations.
- **TALLRec** Bao et al. (2023) fine-tunes LLMs on instruction-style recommendation data.
- **LLaRA** Liao et al. (2024) incorporates user interaction sequences as an additional modality.
- **S-DPO** Chen et al. (2024) extends DPO with softmax and multiple negatives to inject ranking into language models.
- **Rec-SAVER** Tsai et al. (2024) elicits LLM chain-of-thought reasoning for rating prediction and verify its quality without human references.
- **Reason4Rec** Fang et al. (2025) orchestrates LLM reasoning in three deliberative steps—distill, match, predict—supervised by user reviews.
- **RecZero** Kong et al. (2025) trains a single LLM through pure RL to autonomously develop reasoning capabilities for rating prediction.
- **Graphadapter** Huang et al. (2024) uses GNN as an adapter to enhance LLMs' ability to handle text attributes.

A.5 Statistical significance of randomly sampled interaction chains

We conducted significance tests on the random sampling interaction chain results, comparing each sample with the best-performing baseline. The results are shown in Table 6:

Table 6: Statistical significance of randomly sampled interaction chains. HR denotes HitRatio.

Method	MovieLens			Amazon Book			Amazon Electronics		
	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
Best Baseline	0.385	0.512	0.472	0.542	0.631	0.556	0.410	0.563	0.498
R2Rec	0.411	0.549	0.493	0.591	0.664	0.632	0.467	0.602	0.545
p-value	2.46e-4	2.33e-3	5.67e-5	9.48e-7	8.40e-4	8.93e-9	4.32e-7	3.77e-6	1.92e-6

From the table above, it can be observed that the p-value for our method compared to the best-performing baseline is less than 0.01 across all metrics, indicating that our method significantly outperforms the baseline in terms of performance.

A.6 Scalability Analysis

To further investigate the impact of training data size on model performance, we conducted experiments in the RL stage by incrementally increasing the training data from 500 to 1500 training samples. The results are shown in Figure 4.

We observed that as the amount of training data increased, the performance of our model consistently improved. Specifically, when the training data size increased from 500 to 1500, the HitRatio metric on three different datasets showed an improvement of 11.39%. This finding aligns with the results in Guo et al. (2025), which demonstrate that model performance on mathematical tasks improves as the dataset size increases. Similarly, in our recommendation task, the model's performance improved as the training data grew. These results highlight the potential of scaling up training data to enhance the model's ability to capture more complex patterns in user preferences, thereby improving overall recommendation accuracy.

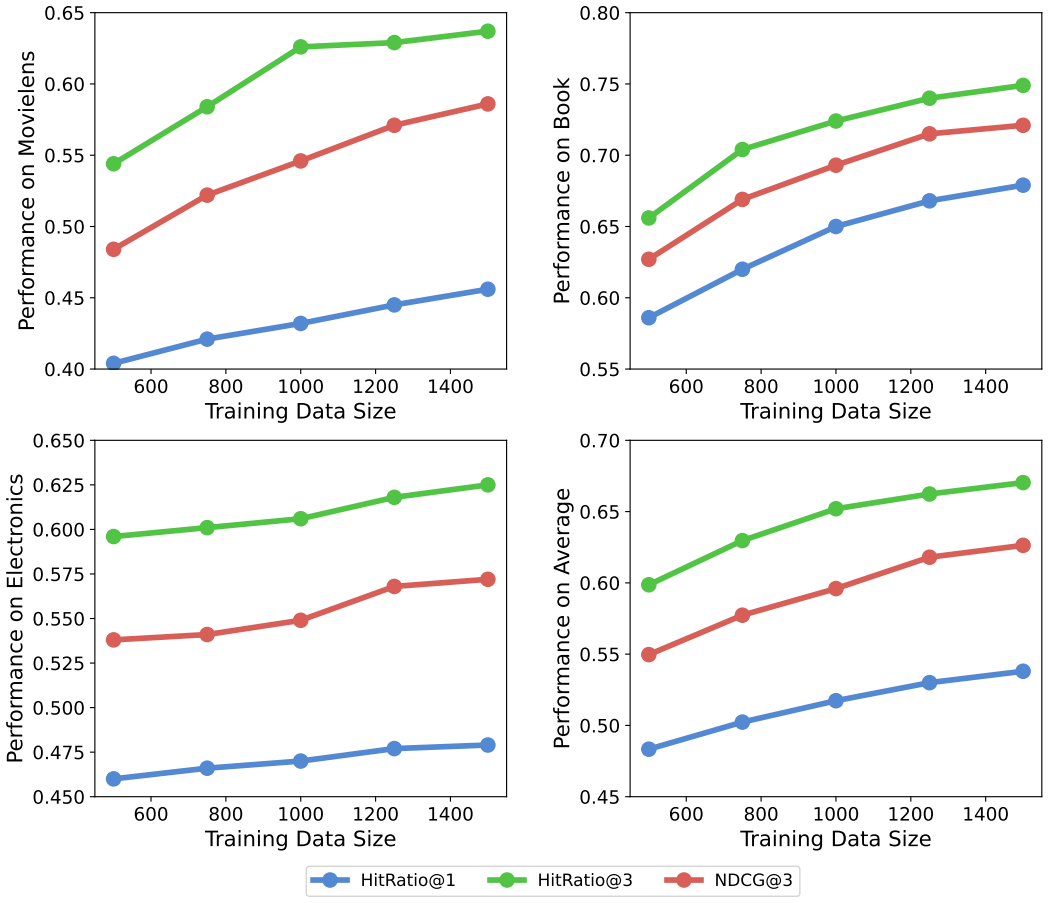


Figure 4: Scaling of Model Performance with Training Data Size

A.7 Performance Analysis in Sparse Scenarios

To evaluate robustness in data-sparse situations, we tested users with limited history (interaction counts ≤ 5). The results in Table 7 show that R2Rec consistently outperforms the baseline. This superiority stems from R2Rec's powerful reasoning capabilities. Even when interaction links are scarce, the model can effectively reason over available semantic information—such as user profiles and item attributes—within the context to deduce latent preferences, thereby compensating for the lack of dense behavioral patterns.

Table 7: Performance comparison on sparse user groups (interaction counts ≤ 5). HR denotes HitRatio.

Method	MovieLens			Amazon Book			Amazon Electronics		
	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
Best Baseline	0.126	0.194	0.178	0.134	0.213	0.177	0.122	0.207	0.185
R2Rec	0.223	0.310	0.281	0.259	0.332	0.316	0.225	0.324	0.287

A.8 Evaluation under Enlarged and Full-Catalog Candidate Spaces

The main experiments follow the widely adopted protocol of ranking a 20-item candidate set. To examine how R2Rec behaves in more demanding evaluation regimes, this appendix first enlarges the candidate set to 100 items (Appendix A.8.1) and then evaluates a full-catalog retrieval-and-reranking pipeline in which candidates are no longer pre-filtered around the ground truth (Appendix A.8.2).

A.8.1 Performance with Increased Candidate Length

To assess the model's discriminative ability, we increased the candidate set size to 100. As shown in Table 8, R2Rec maintains state-of-the-art performance despite the increased noise. This is attributed to the Interaction-of-Thought mechanism, which acts as a precise logical filter. By performing step-by-step reasoning, R2Rec can strictly align user intent with item features, enabling it to distinguish the ground-truth item from a large pool of distractors more effectively than traditional matching methods.

Table 8: Performance comparison with an expanded candidate set size of 100. HR denotes HitRatio.

Method	MovieLens			Amazon Book			Amazon Electronics		
	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
Best Baseline	0.196	0.287	0.248	0.365	0.460	0.439	0.289	0.371	0.338
R2Rec	0.211	0.298	0.260	0.421	0.484	0.455	0.339	0.402	0.376

A.8.2 Full-Catalog Retrieval-and-Reranking Evaluation

Sampled-candidate evaluation, even with 100 candidates, still pre-filters the item space around the ground truth. To evaluate R2Rec under a setting that involves the entire item catalog, we adopt the standard industrial cascade:

$$\text{Full Catalog} \xrightarrow{\text{Retriever}} \text{Top-}K \text{ Candidates} \xrightarrow{\text{R2Rec}} \text{Reranked Top-}K.$$

Concretely, a lightweight retriever first retrieves the top-K ($K = 20$) candidates for each test user from the *full* item catalog, and R2Rec then reranks the retrieved candidates. In this pipeline, R2Rec serves as the reranking stage rather than directly sorting hundreds of thousands of items, which matches how LLM-based rankers are deployed in practice. We instantiate the retriever with two representative models: the sequential recommender SASRec Kang & McAuley (2018) and the generative retrieval model TIGER Rajput et al. (2023). For comparison, the strongest baseline from Table 1 is applied as the reranker on the identical retrieved candidate lists. All metrics are computed over the full test set: if the ground-truth item is not retrieved into the top-K list, the case is counted as a failure for the reranker. Consequently, the end-to-end performance is upper-bounded by the retriever's Recall@20, which we also report. The results are shown in Table 9.

Table 9: Full-catalog retrieval-and-reranking evaluation. The retriever selects the top-20 candidates from the full item catalog, and the reranker reorders them. Users whose ground-truth item is not retrieved are counted as failures, so end-to-end performance is bounded by the retriever's Recall@20. HR denotes HitRatio.

Dataset	Retriever	Recall@20	Best Baseline as Reranker			R2Rec as Reranker		
			HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
MovieLens	SASRec	0.324	0.1268	0.1697	0.1579	**0.1355**	**0.1797**	**0.1612**
	TIGER	0.351	0.1374	0.1823	0.1701	**0.1465**	**0.1996**	**0.1740**
Amazon Book	SASRec	0.052	0.0294	0.0342	0.0320	**0.0303**	**0.0357**	**0.0328**
	TIGER	0.058	0.0317	0.0371	0.0337	**0.0328**	**0.0388**	**0.0362**
Amazon Electronics	SASRec	0.062	0.0260	0.0354	0.0316	**0.0299**	**0.0371**	**0.0336**
	TIGER	0.067	0.0277	0.0380	0.0341	**0.0322**	**0.0413**	**0.0368**

As shown in Table 9, R2Rec consistently outperforms the strongest baseline reranker under both retrievers and on all three datasets, indicating that the advantage of Interaction-of-Thought reasoning observed in the sampled-candidate setting carries over to the full-catalog regime, where the retrieved candidates are hard negatives produced by a real retriever rather than random samples.

A.9 Validity and Faithfulness Analysis of Interaction-of-Thought Traces

Since the Interaction-of-Thought traces serve as the reasoning supervision of R2Rec, we analyze their quality from four perspectives: the necessity of outcome guidance (Appendix A.9.1), step-level faithfulness to interaction evidence (Appendix A.9.2), information leakage (Appendix A.9.3), and the downstream effect of the progressive, masked generation strategy (Appendix A.9.4).

A.9.1 Why Is Outcome-Guided Trace Generation Needed?

A natural question is whether the reasoning traces could be generated *without* providing the observed outcome. To answer this, we keep the interaction chains unchanged but remove the observed answer from the trace-generation prompt, letting the LLM reason freely over the chain. On 100 sampled traces per dataset, we measure the **Outcome-Consistent Trace Ratio**, i.e., the fraction of traces whose final conclusion agrees with the observed endpoint of the closed chain. The results are shown in Table 10.

Table 10: Outcome-Consistent Trace Ratio of answer-free generation versus our outcome-guided generation.

Method	MovieLens	Amazon Book	Amazon Electronics
Answer-free generation	57%	54%	43%
Outcome-guided generation (ours)	**100%**	**100%**	**100%**

Answer-free generation produces a large fraction of outcome-inconsistent supervision: only 43%–57% of the freely generated traces reach a conclusion that agrees with the observed endpoint, and could introduce outcome-inconsistent reasoning signals into SFT supervision. This clarifies the role of the observed answer in our pipeline: it acts as an *outcome-level constraint* that anchors the endpoint of each synthesized trace to real user behavior, rather than as additional interaction evidence.

A.9.2 Step-Level Faithfulness to Interaction Evidence

Outcome consistency alone does not guarantee that the intermediate reasoning is faithful: a trace could reach the correct endpoint through claims unsupported by the interaction chain. We therefore let an independent LLM judge inspect, for 100 sampled traces per dataset, whether each intermediate reasoning step is supported by the interaction-chain evidence visible to the generator at that step, and report the **Evidence-Grounded Rate** (the fraction of supported steps) in Table 11.

Table 11: Step-level Evidence-Grounded Rate of the generated Interaction-of-Thought traces.

Dataset	MovieLens	Amazon Book	Amazon Electronics
Evidence-Grounded Rate	98.50%	97.75%	99.00%

Across all three datasets, the vast majority of intermediate reasoning claims can be supported by the interaction evidence available to the model. Note that outcome consistency and evidence faithfulness are distinct properties: together with Appendix A.9.1, these results show that the synthesized traces are grounded in real interaction evidence while terminating at the observed outcome.

A.9.3 Information Leakage Analysis

When generating reasoning traces via the progressive, masked agentic pipeline, it is crucial to ensure that the synthesized text does not inadvertently reveal the ground-truth labels. To validate the integrity of our pipeline, we analyze the potential for information leakage from two complementary perspectives: (i) demonstrating that the evaluation protocol structurally precludes leakage at inference time, and (ii) quantitatively verifying that the generated reasoning traces do not expose the target answers.

Inference-time isolation of reasoning traces. The reasoning traces produced by the progressive, masked agentic pipeline are used exclusively as supervision during the SFT stage. At inference time, the LLM receives only the user profile, the historical interactions, the sampled interaction chains, and the candidate set; the offline-generated reasoning traces are never fed back to the model. Moreover, the items appearing in the interaction chains do not overlap with the ground-truth item in the candidate set. As a result, no answer-revealing information from the training-time traces can propagate into the evaluation procedure.

Quantitative leakage assessment. To verify that the progressive, masked agentic pipeline itself yields traces that do not prematurely disclose the final answer, we conducted a leakage audit. Specifically, we prompted an independent LLM judge to determine, for each generated reasoning trace, whether the identity of the masked target item could be inferred from the reasoning content alone. We report the non-leakage ratio (Table 12), i.e., the fraction of traces for which the judge determines that the answer is not revealed.

Table 12: Non-leakage ratio of reasoning traces generated by the progressive, masked agentic pipeline.

Dataset	MovieLens	Amazon Book	Amazon Electronics
Non-leakage Ratio	98.6%	98.4%	98.0%

Across all three datasets, more than 98% of the generated traces are judged to be free of answer leakage. The high non-leakage ratios suggest that explicit target information is rarely recoverable from the generated intermediate reasoning traces. We note that this audit addresses only the *explicit* leakage of the target identity; it is complementary to, rather than a substitute for, the step-level evidence faithfulness examined in Appendix A.9.2.

A.9.4 Effect of Trace Generation Strategy

Finally, we compare free trace generation with our trace generation strategy. In the free setting, the observed outcome is omitted and the LLM directly generates the reasoning trace from the interaction chain; all other components remain unchanged. We compare the two strategies at three stages: Prompt, SFT, and Full (SFT+RL). Here, "Ours" denotes our outcome-guided, progressively masked trace generation strategy. The results are shown in Table 13.

Table 13: EEffect of trace generation strategy on downstream performance (HitRatio@1 / HitRatio@3 / NDCG@3).

Stage	Generation	MovieLens	Amazon Book	Amazon Electronics
Prompt	Free	0.217/0.375/0.294	0.351/0.476/0.429	0.181/0.306/0.261
Prompt	Ours	**0.230/0.381/0.317**	**0.366/0.494/0.440**	**0.192/0.326/0.268**
SFT	Free	0.315/0.434/0.386	0.521/0.583/0.559	0.421/0.490/0.448
SFT	Ours	**0.320/0.453/0.397**	**0.528/0.608/0.574**	**0.436/0.490/0.467**
Full	Free	0.394/0.531/0.486	0.577/0.647/0.604	0.450/0.581/0.525
Full	Ours	**0.411/0.549/0.493**	**0.591/0.664/0.632**	**0.467/0.602/0.545**

Our strategy consistently outperforms free generation across all stages and datasets, showing that outcome-guided progressive trace generation provides more effective reasoning supervision for downstream recommendation.

A.10 Generality of R2Rec Across Different Backbone LLMs

To verify that R2Rec is a general framework rather than a recipe tailored to a specific backbone, we replicate the full pipeline (basic prompt $\rightarrow$ + interaction chain $\rightarrow$ + reasoning chain $\rightarrow$ + SFT $\rightarrow$ + RL) on eight backbone LLMs spanning three model families (DeepSeek, LLaMA, Qwen) and a wide range of parameter sizes (1B to 8B). We use the Amazon Book dataset as a representative testbed; trends on the other two datasets are qualitatively consistent. The experimental results are presented in Table 14.

Table 14: Stage-wise performance of R2Rec across eight backbone LLMs on Amazon Book (HitRatio@1 / HitRatio@3 / NDCG@3). Each column corresponds to one cumulative stage of the pipeline.

Model	Basic	+ Interaction Chain	+ Reasoning Chain	+ SFT	+ RL
DeepSeek-R1-8B	0.160/0.324/0.255	0.278/0.420/0.361	0.366/0.494/0.440	0.528/0.608/0.574	**0.591/0.664/0.632**
LLaMA-3.1-8B	0.068/0.104/0.083	0.132/0.186/0.159	0.198/0.296/0.247	0.560/0.636/0.602	**0.579/0.642/0.624**
DeepSeek-R1-7B	0.132/0.278/0.229	0.196/0.326/0.284	0.178/0.296/0.251	0.446/0.518/0.474	**0.563/0.640/0.598**
Qwen2.5-7B	0.130/0.242/0.194	0.162/0.314/0.240	0.252/0.358/0.301	0.494/0.584/0.527	**0.552/0.639/0.604**
Qwen3-4B	0.014/0.052/0.029	0.112/0.196/0.160	0.152/0.224/0.191	0.480/0.538/0.512	**0.522/0.584/0.532**
Qwen3-1.7B	0.038/0.096/0.063	0.074/0.138/0.109	0.080/0.138/0.114	0.436/0.490/0.464	**0.505/0.531/0.519**
LLaMA-3.2-1B	0.006/0.010/0.007	0.014/0.038/0.022	0.028/0.032/0.029	0.286/0.346/0.311	**0.394/0.508/0.473**
LLaMA-3.2-3B	0.016/0.032/0.024	0.092/0.148/0.127	0.196/0.266/0.232	0.460/0.558/0.513	**0.501/0.533/0.510**

The results demonstrate two important properties of R2Rec. First, every backbone benefits monotonically from each successive stage of the pipeline: introducing the interaction chain consistently improves over the basic prompt, augmenting it with the reasoning chain yields further gains, SFT delivers the largest single jump by warm-starting the model's chain-aware reasoning capability, and RL provides additional refinement. Second, R2Rec is effective across a heterogeneous set of model families and across an order of magnitude in parameter scale (1B–8B): regardless of the backbone's initial recommendation ability, the full pipeline elevates all models to a competitive performance level. Together, these observations establish R2Rec as a general framework for eliciting recommendation-oriented reasoning in LLMs, rather than a method specific to any single backbone.

A.11 What Makes a Good Interaction Chain

Beyond evaluating the overall effectiveness of interaction chains, we further investigate which structural properties of a chain contribute most to recommendation performance. We focus on two complementary dimensions: the **composition pattern** of the chain (i.e., the sequence of positive/negative interactions along the chain), and the **hop length** of the chain. The hop-length analysis is reported jointly with the chain-number analysis in Appendix A.12; here we focus on the composition pattern and synthesize the overall design implications.

Effect of chain composition pattern. We let "p" denote a positive interaction and "n" denote a negative one along a chain. For instance, "ppnp" denotes a 3-hop chain in which the target user likes item A, an intermediate user also likes item A but dislikes item B, and the target user has rated item B. We compare the default random sampling strategy (which mixes diverse patterns) with three representative fixed patterns. The experimental results are presented in Table 15.

Table 15: Effect of interaction chain composition pattern. HR denotes HitRatio.

Chain Type	MovieLens			Amazon Book			Amazon Electronics		
	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
pppp	0.383	0.517	0.447	0.545	0.628	0.609	0.421	0.562	0.503
pppn	0.375	0.483	0.469	0.541	0.625	0.590	0.411	0.521	0.473
pnpp	0.379	0.497	0.471	0.537	0.614	0.589	0.415	0.529	0.498
random	**0.411**	**0.549**	**0.493**	**0.591**	**0.664**	**0.632**	**0.467**	**0.602**	**0.545**

Design implications. Combining the composition-pattern results above with the chain-number and chain-length results in Appendix A.12, we summarize four properties that characterize a high-quality interaction chain for LLM-based reasoning:

1. **Multi-hop chains are essential.** A 1-hop chain provides only surface-level content similarity between adjacent items, whereas multi-hop chains expose deeper preference-propagation patterns mediated by intermediate users. The dramatic gap between 1-hop and multi-hop performance (Figure 5) confirms that multi-hop collaborative signals are the principal source of R2Rec's gains.
2. **Moderate chain length is optimal.** Performance peaks around 3-hop chains and degrades noticeably at 7 hops, indicating that excessively long chains introduce noise from distantly related users that dilutes the core collaborative signal. The optimal length balances information richness against signal clarity.
3. **Pattern diversity matters more than any single pattern.** Random sampling consistently outperforms each fixed composition (pppp, pppn, pnpp). This suggests that exposing the model to a heterogeneous mix of agreement, disagreement, and contrastive chains is more informative than overfitting to one pattern, since different patterns capture complementary facets of preference dynamics.
4. **Positive-agreement chains are the strongest individual pattern.** Among fixed patterns, "pppp" consistently yields the best results, indicating that consistent positive agreement between the target user and intermediate users provides a particularly informative collaborative signal.

These findings provide concrete guidance for constructing interaction chains in future graph-based reasoning frameworks for recommendation.

A.12 Analysis of the Impact of Interaction Chain Number and Length on Model Performance

Two structural hyperparameters jointly determine how much collaborative evidence a sampled interaction chain provides to the LLM: the number of chains presented per target user, and the length (number of hops) of each chain. The former controls the breadth of the collaborative neighborhood that the model is exposed to, while the latter controls the depth of preference propagation captured along each chain. We analyze both factors below.

Effect of the number of interaction chains. we analyzed changes in HitRatio@1 with varying numbers of chains, as shown in the left panel of Figure 5. The results demonstrate a clear improvement in HitRatio@1 as the number of chains increases, indicating that more interaction chains help the model better capture user preferences, enhancing both reasoning and recommendation accuracy. Specifically, a larger number of chains allows the model to integrate more interactions, improving its understanding of user behavior and, ultimately, recommendation quality.

Effect of interaction chain length. We then vary the chain length from 1-hop to 7-hop while keeping all other components fixed, as shown in the right panel of 5. Two clear trends emerge.

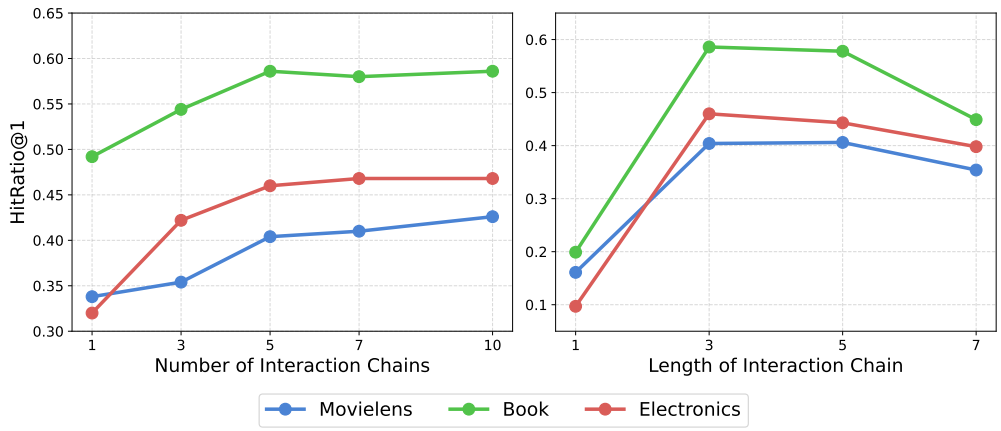


Figure 5: Effect of interaction chain number on model performance.

First, moving from 1-hop to 3-hop chains brings dramatic gains across all datasets, confirming that genuinely multi-hop collaborative evidence is what drives the effectiveness of R2Rec, not merely the inclusion of a single neighboring interaction. Second, performance largely saturates at 3 hops, with 5-hop chains offering only marginal differences and 7-hop chains causing a clear regression. We attribute the degradation at 7 hops to two factors: longer chains incorporate more distant intermediate users whose preferences are less aligned with the target user, diluting the relevant signal; and the increased reasoning length raises the difficulty of extracting a coherent collaborative pattern.

A.13 Analysis of RL Training and Reward Design

This appendix analyzes the reinforcement learning stage from two aspects: the reward dynamics during training (Appendix A.13.1) and the contribution of each reward component (Appendix A.13.2).

A.13.1 Reward Curves during RL Training

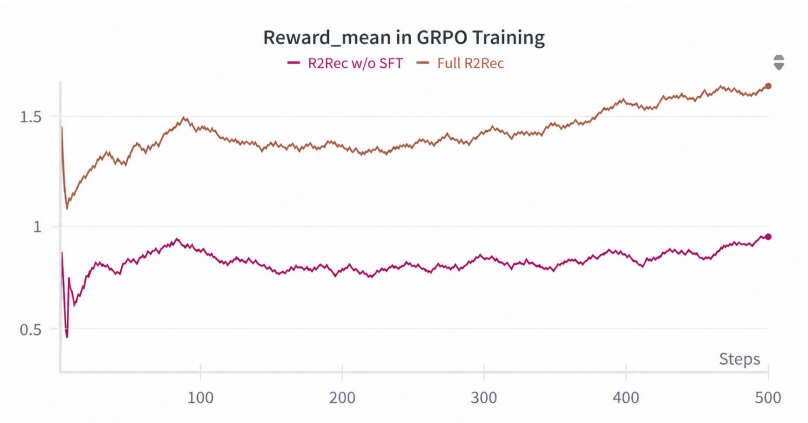
To better understand the contribution of supervised fine-tuning (SFT) in initializing the LLM's reasoning capabilities, we analyze the reward trajectories during reinforcement learning (RL) training for models with and without SFT initialization.

As shown in Figure 6, the LLM initialized with SFT achieves a significantly higher starting reward and maintains stable reward growth throughout the training process. This indicates that SFT equips the model with a fundamental ability to reason over user–item interaction chains, allowing it to benefit more effectively from reward signals during RL.

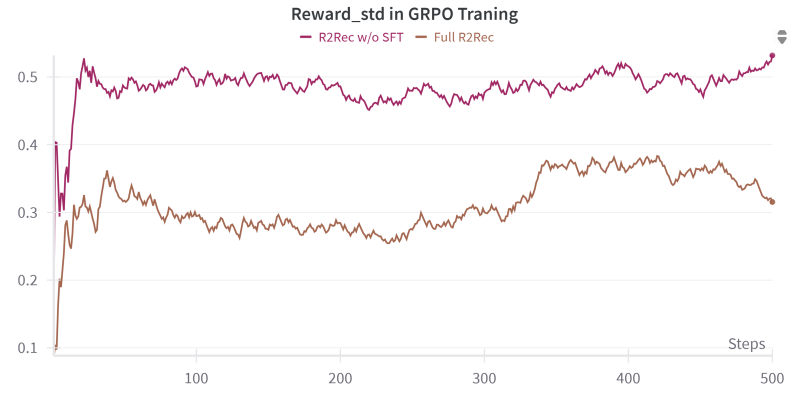
In contrast, the model trained directly with RL from scratch exhibits lower and more volatile rewards. It converges more slowly and fails to reach the performance of the SFT-initialized counterpart. These findings highlight that supervised fine-tuning is a critical component for stabilizing RL training, especially in domains lacking abundant reasoning supervision, as it provides the model with essential prior structure to build upon during reward-driven optimization.

A.13.2 Reward Component Ablation

The total reward in the RL stage combines a reasoning length reward and a ranking correctness reward. To isolate the contribution of each component, we retrain the RL stage with only one of the two rewards enabled, keeping the SFT initialization, the training data, and all other hyperparameters identical. The results are shown in Table 16.



(a) The mean value of the reward during GRPO Training



(b) The variance of the reward during GRPO Training

Figure 6: Reward comparison between RL training with and without SFT initialization.

Table 16: Ablation of the reward components in the RL stage. HR denotes HitRatio.

Reward	MovieLens			Amazon Book			Amazon Electronics		
	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
Length only	0.341	0.483	0.436	0.537	0.622	0.591	0.439	0.522	0.486
Correctness only	0.350	0.498	0.454	0.551	0.624	0.589	0.445	0.537	0.493
Full (length + correctness)	**0.411**	**0.549**	**0.493**	**0.591**	**0.664**	**0.632**	**0.467**	**0.602**	**0.545**

The full reward achieves the best performance on all metrics across the three datasets. The correctness-only reward is slightly stronger than the length-only reward on HitRatio@1, as it directly optimizes the ranking objective, while the length-only reward alone still clearly improves over the SFT-only model, indicating that encouraging a sufficient decomposition of the reasoning process is itself a useful training signal. Combining the two rewards outperforms either component alone, supporting the design of the total reward in Section 3.3.

A.14 Comparison with Classical Baselines under Cross-Domain Transfer

In Section 4.4 we report the cross-domain transferability of R2Rec to demonstrate that the model acquires a transferable reasoning capability over interaction chains, rather than memorizing domain-specific user–item patterns. To better contextualize this result, we further compare R2Rec against the strongest *trainable classical* baselines under the same cross-domain transfer protocol. For each

(training domain, test domain) pair, we report the result of the best-performing classical baseline. The experimental results are presented in Table 17.

Table 17: Cross-domain transfer performance of the best classical baseline. Each row indicates the training domain; each column indicates the test domain. HR denotes HitRatio.

Training → Test	MovieLens			Amazon Book			Amazon Electronics		
	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3	HR@1	HR@3	NDCG@3
MovieLens	0.385	0.512	0.472	0.356	0.474	0.431	0.179	0.264	0.233
Amazon Book	0.201	0.331	0.302	0.542	0.631	0.556	0.285	0.351	0.308
Amazon Electronics	0.146	0.227	0.185	0.323	0.405	0.371	0.410	0.563	0.498

Comparing this table with the cross-domain results of R2Rec (Table 3 in the main paper), R2Rec consistently outperforms the best classical baseline across every off-diagonal (out-of-domain) cell. The performance gap is particularly pronounced when the training and test domains differ in modality and item space (e.g., MovieLens → Amazon Electronics): classical sequential models suffer severe degradation, whereas R2Rec retains substantially stronger generalization. This supports the conclusion that R2Rec internalizes a domain-agnostic reasoning capability over the structural patterns of interaction chains, rather than memorizing surface-level user–item co-occurrence statistics specific to a single domain.

A.15 Details settings of user study

In this study, we evaluate the effectiveness of our proposed method compared to baseline approaches using a small sample set consisting of ten data samples, randomly selected from the MovieLens, Book, and Electronics datasets. To assess the quality of the reasoning processes generated by these methods, we invited 34 participants to rate the generated reasoning across various dimensions on a 5-point scale. The evaluation criteria were designed to capture different aspects of the reasoning quality and are as follows:

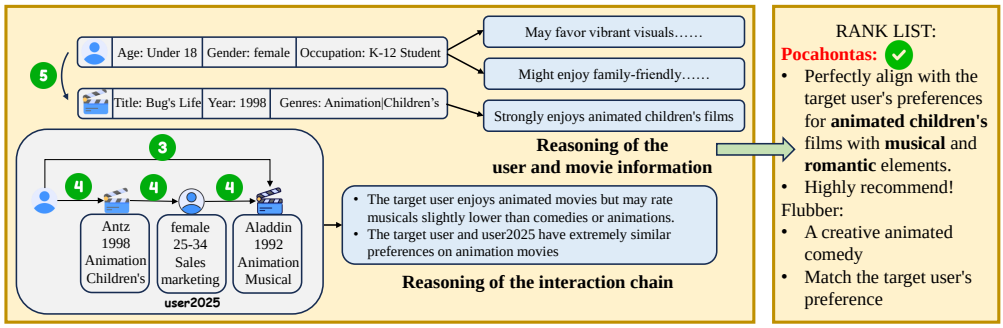


Figure 7: Case Study of the reasoning process generated by R2Rec.

- **Relevance:** How relevant is the information used in the reasoning process to the target user's data?
- **Interpretability:** How effectively does the reasoning process utilize the user's data to make the final recommendation?
- **Overall Quality:** A general assessment of the quality of the reasoning process, considering the clarity, coherence, and usefulness of the generated explanation.

To provide a more intuitive demonstration of the quality and interpretability of the reasoning process in our approach, we present a case study involving a female user under 18 years old, as shown in Figure 7. Based on her demographic profile and a high rating for A Bug's Life (1998), the model infers a preference for animated, children-oriented comedies. Reasoning over an interaction chain involving shared ratings of Antz and Aladdin with a similar user further refines this understanding, suggesting a nuanced preference for animation and lighthearted themes over musicals. Given this profile, the model ranks Pocahontas (1995)—an animated musical with children's and romantic

elements—as the most suitable recommendation. This example highlights how our method enables interpretable recommendations by explicitly tracing the reasoning from both personal history and interaction structure.

A.16 Compute resources

All experiments were conducted using NVIDIA A100-80G GPUs. For supervised fine-tuning (LoRA-based SFT), we used a single A100 GPU, with each training run taking approximately 2.5 hours. For reinforcement learning with the GRPO algorithm, we utilized two A100 GPUs, and each training run took approximately 8 hours. Evaluation and inference tasks were conducted using a single A100 GPU.

Table 18 presents the comparison of inference time between our method and other reasoning-enhanced baselines. The results indicate that R2Rec maintains acceptable latency while ensuring superior recommendation performance and high interpretability. This demonstrates that our framework strikes a favorable balance between efficiency and effectiveness, making it practical for real-world deployment.

Table 18: Inference time comparison between R2Rec and other reasoning-enhanced methods.

Method	MovieLens	Amazon Book	Amazon Electronics
Rec-SAVER	0.64s	0.89s	1.07s
Reason4Rec	-	0.95s	1.13s
RecZero	0.71s	1.02s	1.27s
R2Rec	0.76s	1.08s	1.24s

A.17 Open-Source Frameworks Used in This Work

The implementation of R2Rec builds on three open-source frameworks that we gratefully acknowledge:

- **LLaMA-Factory**[1]: We use LLaMA-Factory to conduct the supervised fine-tuning (SFT) stage of R2Rec. It provides a unified, efficient interface for LoRA-based fine-tuning across a wide range of LLM backbones.
- **Open-R1**[2]: We adopt the Open-R1 codebase to implement the GRPO-based reinforcement learning stage of R2Rec. Open-R1 provides a faithful and well-engineered implementation of GRPO that we adapt to our recommendation-specific reward design.
- **vLLM**[3]: We use vLLM as the inference backend for both reasoning trace generation during data preparation and model evaluation. Its high-throughput serving substantially accelerates large-scale inference over the candidate-set ranking task.

We thank the developers and maintainers of these projects, whose contributions to the open-source community made this work possible.

A.18 Limitations

Despite the superior performance of R2Rec, one limitation lies in the static nature of interaction chain construction. In our current framework, interaction chains are pre-constructed based on historical graph data rather than being dynamically generated during the inference phase. While this approach ensures training stability and efficiency, it may limit the model's agility in adapting to real-time, dynamic shifts in user preferences. Future work could explore dynamic graph traversal mechanisms to construct reasoning chains on-the-fly, thereby enhancing the system's responsiveness to immediate user feedback.

[1] https://github.com/hiyouga/LLaMA-Factory
[2] https://github.com/huggingface/open-r1
[3] https://github.com/vllm-project/vllm

A.19 BROADER IMPACTS

This work advances recommendation systems by utilizing Interaction-of-Thought reasoning traces, improving both accuracy and transparency. On the positive side, our approach makes LLM-based recommendations more interpretable and trustworthy, while our two-stage training pipeline (SFT+RL) offers a cost-effective paradigm for eliciting complex reasoning. However, there are potential negative societal impacts. The enhanced recommendation precision could inadvertently exacerbate "filter bubbles," limiting users' exposure to diverse content. Additionally, the model risks inheriting and amplifying historical biases or demographic stereotypes present in the interaction graphs and pre-trained LLMs. Future deployments should mitigate these risks by incorporating fairness constraints and diversity-promoting reward signals.