LLMRG: Improving Recommendations through Large Language Model Reasoning Graphs

Yan Wang[1*], Zhixuan Chu[1*†], Xin Ouyang[1], Simeng Wang[1], Hongyan Hao[1], Yue Shen[1], Jinjie Gu[1], Siqiao Xue[1], James Zhang[1], Qing Cui[1], Longfei Li[1], Jun Zhou[1], Sheng Li[2]

[1]Ant Group
[2]University of Virginia
{luli.wy,chuzhixuan.czx,xin.oyx,simeng.wsm,hongyanhao.hhy,zhanying,jinjie.gujj,siqiao.xsq,james.z,cuiqing.cq,longyao.llf, jun.zhoujun}@antgroup.com, shengli@virginia.edu

Abstract

Recommendation systems aim to provide users with relevant suggestions, but often lack interpretability and fail to capture higher-level semantic relationships between user behaviors and profiles. In this paper, we propose a novel approach that leverages large language models (LLMs) to construct personalized reasoning graphs. These graphs link a user's profile and behavioral sequences through causal and logical inferences, representing the user's interests in an interpretable way. Our approach, LLM reasoning graphs (LLMRG), has four components: chained graph reasoning, divergent extension, self-verification and scoring, and knowledge base self-improvement. The resulting reasoning graph is encoded using graph neural networks, which serves as additional input to improve conventional recommender systems, without requiring extra user or item information. Our approach demonstrates how LLMs can enable more logical and interpretable recommender systems through personalized reasoning graphs. LLMRG allows recommendations to benefit from both engineered recommendation systems and LLM-derived reasoning graphs. We demonstrate the effectiveness of LLMRG on benchmarks and real-world scenarios in enhancing base recommendation models.

Introduction

Recommendation systems are now prevalent across the internet, smartly surfacing personalized content and products to users based on their individual profiles and historical behavioral data. However, most recommendation systems rely solely on conventional machine learning techniques, which can only identify patterns and relationships within sequences of interactions without actually comprehending the true meaning or semantics behind the items themselves. Devoid of any logical or causal reasoning capacities (Li and Chu 2023; Chu, Rathbun, and Li 2021), these recommender systems struggle to effectively capture the full spectrum of conceptual relationships and connections spanning a user's diverse interests and behavioral patterns over time. In addition, recent work (Wang et al. 2019; Chen et al. 2021; Wu et al. 2019; Wang et al. 2020; Sheu et al. 2021) has sought to enhance recommendations by incorporating graph-structured information, which provides valuable contextual data beyond standard tabular data formats. However, even these more advanced knowledge graphs used in recommendation systems still lack the ability to perform complex reasoning or inference - simply overlaying factual relationships is not enough to enable a system to deeply understand users' interests and generate insightful recommendations.

In parallel, tremendous progress in large language models (LLMs) has demonstrated powerful new capacities for reasoning, inference, and logic without the need for explicit training on such tasks. These models exhibit remarkable aptitudes for causal, logical, and analogical reasoning, illuminating new opportunities to leverage their strengths to develop superior knowledge representations that can capture nuanced semantic relationships between users' interests. By leveraging LLMs to reason behavioral sequences and comprehend user interests at a deeper conceptual level, there is immense potential to revolutionize next-generation recommendation systems (Chu et al. 2023b; Wang et al. 2023).

Therefore, we propose using an LLM to construct personalized reasoning graphs for recommendation systems. The LLM inputs a user's profile and behavioral sequences and outputs a graphical representation linking concepts through chained causal and logical reasoning. This results in an expansive graph embedding higher-level semantic relationships between the user's interests and behaviors. We then apply graph neural networks to learn a dense feature representation that summarizes the graph's structure and meaning. This graph embedding is provided as additional input to a conventional recommendation model. Our approach allows recommendations to consider conceptual relationships derived through reasoning while still benefiting from the recommendation abilities of traditional models. Moreover, the graph provides interpretability by surfacing the explicit reasoning behind recommendations.

We designed four interlocking modules, powered by large language models (LLMs), to construct personalized reasoning graphs that model each user's interests: 1) a chained graph reasoning module that conducts chained causal and logical reasoning, 2) a divergent extension module that expands the graph by associating and reasoning about the user's interests, 3) a self-verification and scoring module that

*These authors contributed equally.
†Corresponding author.

validates the reasoning procedure through abductive reasoning and scoring, and 4) a knowledge base self-improving module that caches validated reasoning chains for later reuse. Together, these four modules construct the Large Language Model Reasoning Graphs (LLMRG) paradigm, which employs a prompt-based framework leveraging large language models to imaginatively generate plausible new reasoning chains, given their behavioral history and features. Besides, it can perform imaginary continuations of each reasoning chain to predict the next items the user is likely to engage with. This divergent thinking allows us to go beyond reactive recommendations based on consumed content to proactively recommend new items tailored to modeling the user's motivations. Experiments demonstrate our model's ability to improve recommendation performance without requiring additional user or item data. This work illustrates how large language models can enable logical and interpretable recommender systems.

Background

Graph-based Recommendation System

Recent work explores graph-based methods that can incorporate additional relationship information into recommendation systems (Sheu et al. 2021; Chu et al. 2024). For example, knowledge graphs have emerged as a powerful way to represent relationships between entities to capture complex entity interactions (Wang et al. 2019; Chen et al. 2021). Beyond predefined knowledge graphs, some methods (Wang et al. 2019) learn to construct an informative graph from user-item interactions. While knowledge graphs provide external information, graph learning methods (Wu et al. 2019) can extract latent structures. Combining the two concepts, (Wang et al. 2020) jointly leverage a knowledge graph and interaction graph. In summary, graph-based methods allow recommendation models to encode richer connectivity patterns. However, there are some potential disadvantages of graph-based recommendation systems compared to reasoning graph construction by large language models (LLMs): (1) Knowledge graphs require extensive human expertise to build and maintain relationships, whereas LLMs can automatically extract relational knowledge from large text corpora; (2) Predefined knowledge graphs may have coverage gaps for certain entities or domains. LLMs can learn to reason about any entity mentioned in the text; (3) Graph learning methods that construct graphs from interactions are limited to observable user-item connections. LLMs can infer more abstract and latent relationships through reasoning; (4) Knowledge graphs and graphs are static after construction. LLMs can continue to expand their knowledge and reasoning capabilities as they are trained on more data.

Reasoning of LLM

Recent advances in large language models (LLMs) like GPT-3 and PaLM have enabled strong capabilities in logical and causal reasoning (Chu et al. 2023a; Guan et al. 2023; Xue et al. 2023). This progress stems from three key strengths. First, natural language understanding allows LLMs to parse meaning and relationships from text (Devlin et al. 2018; Brown et al. 2020). Models can identify entities, actions, and causal chains through techniques like self-attention and contextual embeddings. Second, LLMs have accumulated vast commonsense knowledge about how the world works (Shin et al. 2021; Chowdhery et al. 2022). GPT-3 was trained on over a trillion words from the internet, absorbing implicit knowledge about physics, psychology, and reasoning. Models like PaLM were further trained with constrained tuning to better incorporate common sense. This enables filling in missing premises and making deductions. Third, transformer architectures impart combinatorial generalization and symbolic reasoning abilities (Wei et al. 2022). Self-attention layers allow LLMs to chain ideas, follow arguments step-by-step, and make coherent deductions. Together, these strengths of understanding language, leveraging knowledge, and combinatorial reasoning empower LLMs to parse scenarios, tap relevant knowledge, and reason through implications and causes.

LLMRG

Problem Statement

Let $\mathcal{U}=\{u_1, u_2, \ldots, u_{|\mathcal{U}|}\}$ denote the set of users, $\mathcal{V}=\{v_1, v_2, \ldots, v_{|\mathcal{V}|}\}$ be the set of items, and list $\mathcal{S}_u=[v_1^{(u)}, \ldots, v_t^{(u)}, \ldots, v_{n_u}^{(u)}]$ denote the sequence of interactions for user $u \in \mathcal{U}$ in chronological order, where $v_t^{(u)} \in \mathcal{V}$ is the item interacted with at time step t and n_u is the length of the sequence. We use relative time indices instead of absolute timestamps. In addition, let $\mathcal{A}_u=[a_1^{(u)}, \ldots, a_i^{(u)}, \ldots, a_{n_a}^{(u)}]$ represent user attributes for modeling personalization, where n_a is the number of attributes. Given a user's interaction history $\mathcal{S}_u$, the sequential recommendation task is to predict the item user u will interact with at the next time step $n_u + 1$.

In this work, we propose constructing Large Language Model Reasoning Graphs (LLMRG), a new paradigm that utilizes LLMs to improve recommendation system performance. We first use a large language model (LLM) to construct personalized reasoning graphs based on $\mathcal{S}_u$ and $\mathcal{A}_u$, which reason a user's profile and behavioral sequences through causal and logical inferences. The graph provides an interpretable model of a user's interests and embeds rich semantic relationships. We propose an adaptive reasoning architecture with self-verification based on the capabilities of LLMs, which includes four components: 1) chained graph reasoning, 2) divergent extension, 3) self-verification and scoring, and 4)a self-improved knowledge base. By encoding the resulting conceptual reasoning graph using graph neural networks, it can be provided as an additional input into conventional recommender systems. This allows recommendations to benefit from both engineered recommendation algorithms and the explanatory knowledge derived from the LLM graph reasoning process.

Adaptive Reasoning Architecture

Chained Graph Reasoning. Along with the user behavioral sequences $\mathcal{S}_u$, for each item, we construct reasoning chains RC_n that link it to existing chains if there are logical

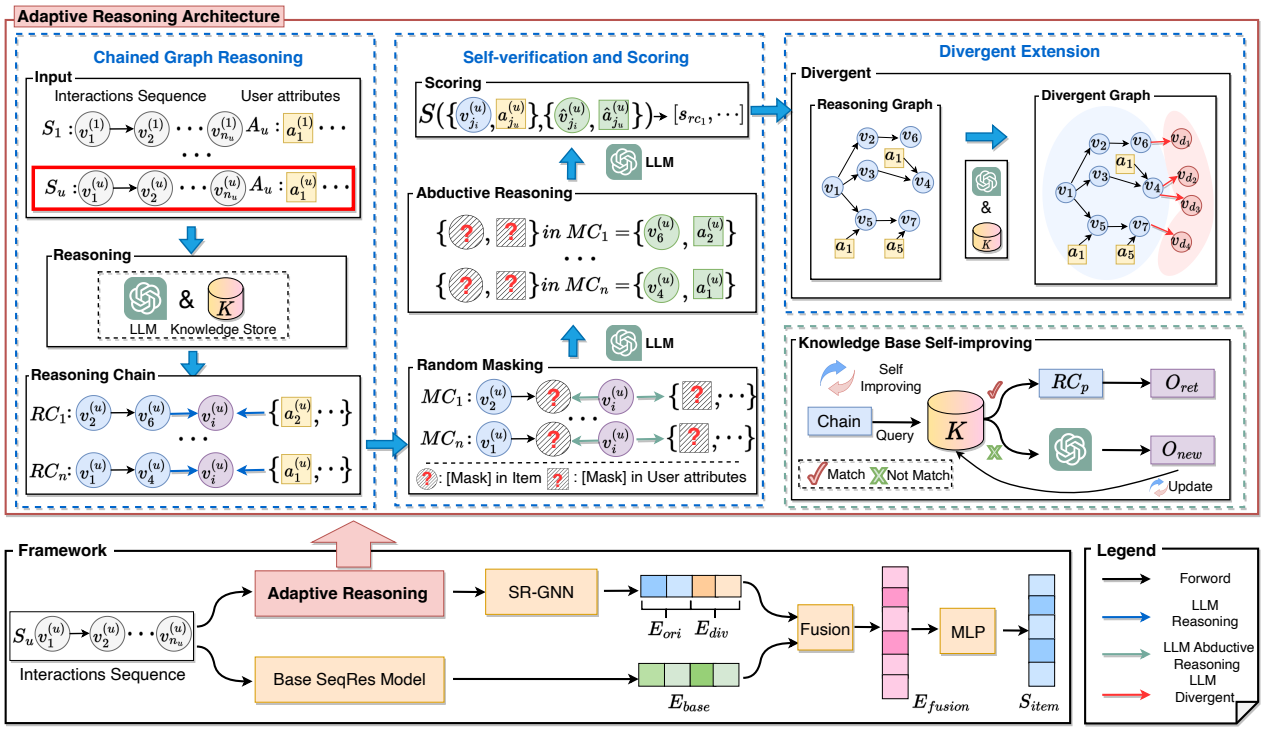

Figure 1: LLMRG framework has two main components, i.e., an adaptive reasoning module with self-verification and a base sequential recommendation model. Our model concatenates the embeddings from the adaptive reasoning module (E_{ori} and E_{div}) and the base model (E_{base}) to obtain E_{fusion}. This fused embedding is used to predict the next item for the user.

connections or start entirely new chains rooted in the item itself if there are no applicable links to existing reasoning chains. Relevant user attributes $\mathcal{A}_u$ are incorporated where possible to further customize the reasoning chains for recommending items. This iterative reasoning chain construction process is carried out progressively along the user's behavioral sequence up until the last item. Specifically, we employ a prompt-based framework leveraging large language models to imaginatively generate plausible new reasoning chains that could logically motivate the user to engage with the next known item in their sequences. The prompt takes as input the known next item, existing reasoning chains constructed thus far, and available user attributes. It outputs a comprehensive set of possible new reasoning chains explaining why the user might want to take the next item. These dynamically generated new chains are integrated into the evolving logical reasoning graph to enable the modeling of increasingly complex interdependent motivations and interests underlying the user's evolving behavioral trajectory.

Divergent Extension. Besides the observed behavioral sequences, we aim to conduct divergent thinking according to the established reasoning graph. We propose a new divergent extension module that performs imaginary continuations of each reasoning chain to predict the next items the user is likely to engage with. Specifically, for each reasoning chain digging into the user's motivations and thinking process, the divergent extension module employs an imagination engine to divergently extend the chain beyond the last known item. This involves using the language model to sample plausible continuations of the reasoning trajectory that predict what other related items the user might be interested in next. For example, if the chain represents an interest in sci-fi movies with complex philosophies, the extension could generate new sequences predicting more cerebral sci-fi films with similar themes and tones that the user might enjoy. Critically, the imagination engine outputs multiple diverse possible extending items per reasoning chain, capturing the user's multifaceted interests. These imaginary new items represent predictions of movies the user is likely to watch soon. We aggregate the predicted new items from all the extended reasoning chains to form a comprehensive set of personalized recommendations tailored to the user's preferences. It is worth noting that the generated new item recommendations may not exist in the original item list for our recommendation task. Therefore, we need to use another small language model to calculate the similarity between the generated items and the original list in order to retrieve the most relevant item recommendations. Divergent thinking allows us to go beyond reactive recommendations based on consumed content to proactively recommend new items tailored to modeling the user's motivations. In this procedure, a prompt-based framework based on LLM is still employed. It is worth noting that rather than just predicting the single next movie, our divergent extension module enables the generalization of multiple future trajectories per reasoning chain. This allows for properly capturing the user's diverse inter-

ests and possibilities they may take next.

Self-verification and Scoring. The self-verification module utilizes the abductive reasoning capability (Xu et al. 2023) of LLM to check the plausibility and coherence of the dynamically generated reasoning chains from the chained graph reasoning and divergent extension modules. Before adding a new reasoning chain to the graph, the module masks the key items or engaged user attributes that the chain is meant to logically link to. It then prompts the large language model to fill in the [Mask] in the masked chains $MC_n^{(u)}$ with the most reasonable prediction. If the predicted item or attribute matches what was originally masked, this provides evidence that the reasoning chain logically flows and is consistent with the user's behavioral history and attributes. The higher the match score, the more robust the reasoning graph is as a whole. On the other hand, a low match score indicates potential flaws in the coherence or plausibility of some reasoning chains. The system can then selectively filter out or recalibrate the problematic chains before integrating them into the graph. Therefore, we set a threshold score for this self-verification to judge the rationality of reasoning. This improves the overall soundness of the dynamically constructed reasoning chains for the chained graph reasoning and divergent extension modules, ensuring reliable reasoning for recommendations aligned with the user's interests. Specifically, this module mainly involves three steps, i.e., random masking, abductive reasoning, and scoring, which are exemplified in Figure 1.

Knowledge Base Self-improving. In our system's chained graph reasoning, divergent extension, and self-verification modules, we make extensive use of a language model to conduct inference and reasoning. This repeated language model invocation incurs significant computational costs. However, we observed that many knowledge elements and reasoning procedures are applied repeatedly across queries. To avoid redundant work, we introduce a knowledge base that caches validated reasoning chains for later reuse. By reusing previous reasoning results rather than re-computing them, we substantially reduce language model usage. We employ a self-improving approach to maintain knowledge base quality over time. Using the scores from our self-verification and scoring module, which assess reasoning chain validity, we retain only high-quality chains in the knowledge base. Low-scoring chains are discarded to filter out low-quality or erroneous inferences. Before conducting new reasoning, we first check whether the knowledge base already contains a relevant chain. If so, we retrieve and leverage that pre-computed chain instead of invoking the language model. This knowledge base of cached, high-quality reasoning chains significantly reduces computational requirements. Our experiments demonstrate it can cut language model usage by about 30% compared to inferences from scratch after 3000 times of reasoning and verification steps in Figure 4.

LLMRG Framework

Sequential recommendation approaches typically view the user's history of interactions as an ordered sequence and attempt to model the user's dynamically evolving interests. In this work, we propose to use an LLM to construct personalized reasoning graphs for recommendation systems. Therefore, as shown in Figure 1, our proposed LLMRG has two components, i.e., an adaptive reasoning module with self-verification and a base sequential recommendation model.

The adaptive reasoning module takes the user's interaction sequence $\mathcal{S}_u$=$[v_1^{(u)}, \ldots, v_t^{(u)}, \ldots, v_{n_u}^{(u)}]$ and attributes $\mathcal{A}_u$=$[a_1^{(u)}, \ldots, a_i^{(u)}, \ldots, a_{n_a}^{(u)}]$ as input. This input goes through chained graph reasoning, self-verification and scoring, and divergent extension repeatedly to construct a reasoning graph and a divergent graph. The adaptive reasoning module is expressed as a mapping $\phi : \{\mathcal{S}_u, \mathcal{A}_u\} \rightarrow \{G_{rea}, G_{div}\}$, where G_{rea} and G_{div} represent the reasoning graph and divergent graph, respectively. We utilize SR-GNN (Wu et al. 2019) to automatically extract embeddings from the graphs, considering the rich node connections. This process produces two embeddings E_{ori} for the reasoning graph and E_{div} for the divergent graph by $g_1 : G_{rea} \rightarrow E_{ori}$ and $g_2 : G_{div} \rightarrow E_{div}$. In parallel, the base sequential recommendation model directly processes the input to produce an embedding E_{base}. Finally, we concatenate the embeddings from the adaptive reasoning module (E_{ori} and E_{div}) and the base model (E_{base}) to obtain E_{fusion}. This fused embedding is used to predict the next item for the user by $\psi : E_{fusion} \rightarrow v_{n_u+1}^{(u)}$.

The key advantages of our approach are that the adaptive reasoning module can construct personalized reasoning graphs, and the divergent extension employs divergent thinking to go beyond reactive recommendations to proactively recommend new items following the evolving behavioral trajectory. The self-verification and scoring also help improve the reasoning process. Fusing this with a standard sequential recommendation model allows for combining complementary strengths without accessing extra information.

Experiments

Settings. To evaluate our proposed method, we conduct experiments on three benchmark datasets: the Amazon Beauty, Amazon-Clothing, and MovieLens-1M (ML-1M) datasets (McAuley et al. 2015; Harper and Konstan 2015). The statistics of the three datasets after preprocessing are summarized in Table 1. To evaluate the performance of our recommendation system, we utilize a leave-one-out strategy where we repeatedly hold out one item from each user's sequence of interactions. We report two widely used ranking metrics - Top-n metrics **HR@**n (Hit Rate) and **NDCG@**n (Normalized Discounted Cumulative Gain) where n is set to 5 and 10. HR@n measures whether the held-out item is present in the top-n recommendations, while NDCG@n considers the position of the held-out item by assigning higher scores to hits at the top ranks. Following the experiment comparison (Du et al. 2023), we include four baseline methods: BERT4Rec (Sun et al. 2019) adopts a bidirectional Transformer as the sequence encoder. FDSA (Zhang et al. 2019) applies self-attention blocks to capture transition patterns of items and attributes. CL4SRec (Xie et al. 2022) proposes data augmentation strategies for contrastive learn-

Specs.	Beauty	Clothing	ML-1M
# Users	22,363	39,387	6,041
# Items	12,101	23,033	3,417
# Avg.Length	8.9	7.1	165.5
# Actions	198,502	278,677	999,611
Sparsity	99.93%	99.97%	95.16%

Table 1: Statistics of the datasets after preprocessing.

Dataset	Metric	FDSA	BERT4Rec	CL4SRec	DuoRec
ML-1M	HR@5	0.0909	0.1124	0.1141	0.2011
	HR@10	0.1631	0.1910	0.1866	0.2837
	ND@5	0.0599	0.0713	0.0721	0.1265
	ND@10	0.0878	0.0980	0.1013	0.1663
Amazon Beauty	HR@5	0.0237	0.0201	0.0398	0.0552
	HR@10	0.0418	0.0413	0.0664	0.0839
	ND@5	0.0195	0.0192	0.0221	0.0350
	ND@10	0.0275	0.0263	0.0322	0.0447
Amazon Clothing	HR@5	0.0119	0.0128	0.0166	0.0190
	HR@10	0.0197	0.0202	0.0273	0.0311
	ND@5	0.0073	0.0081	0.0093	0.0118
	ND@10	0.0109	0.0113	0.0125	0.0155

Table 2: Performance comparison of baseline models on three benchmark datasets. Higher is better.

ing in the sequential recommendation. DuoRec (Qiu et al. 2022) proposes both supervised and unsupervised sampling strategies for contrastive learning in the sequential recommendation.

Results and Analysis. As evidenced in Table 2 and 3, we conduct comprehensive benchmarking experiments on three widely-used datasets - ML-1M, Amazon Beauty, and Amazon Clothing. We compare our proposed LLMRG model built on top of GPT3.5 or GPT4 with several strong baseline methods, including FDSA (Zhang et al. 2019), BERT4Rec (Sun et al. 2019), CL4SREC (Xie et al. 2022), and DuoRec (Qiu et al. 2022). We observe significant performance gains on HR@5, HR@10, NDCG@5, and NDCG@10 after applying LLMRG, compared to the original baseline models. This indicates that conventional recommender systems struggle to model the conceptual relationships and behavioral sequences of diverse user interests. In contrast, our proposed LLMRG framework can boost recommendation performance without needing any additional information. These improvements showcase how large language models can bring logical reasoning and interpretability to recommender systems. Furthermore, LLMRG performance scales with the underlying LLM capability - the GPT4-based LLMRG consistently outperforms its GPT3.5 counterpart. In addition, when comparing the ML-1M movie dataset to the Beauty and Clothing product datasets, we observed that our LLMRG approach led to greater improvements across all evaluation metrics on the ML-1M dataset. This suggests that movie items contain richer semantic information and enable more semantically logical reasoning relationships than Amazon product items. As movies often have complex plots, character arcs, and artistic themes, recommending movies likely requires more sophisticated relational reasoning between items than recommending simple retail products. The complexity of logical relations between movie entities enables our LLMRG method to better leverage its relational modeling capabilities. In contrast, beauty and clothing products have less narrative complexity, so there is less opportunity for relational reasoning to improve recommendations.

Ablation Study. To demonstrate the effectiveness of our proposed reasoning graph, we conduct ablation studies on our LLMRG model using two benchmark datasets: ML-1M and Amazon Beauty. We compare LLMRG to the DuoRec baseline model as well as DuoRec augmented with a simple sequence graph, as proposed by Wu et al. (2019). The sequence graph directly models interaction sequences without reasoning. We also compare against combining DuoRec with large language models - GPT-3.5 and GPT-4 - without constructing a reasoning graph. Here, the LLM simply outputs recommended items based on prompts containing historical sequences and user profiles, without a reasoning graph. As shown in Table 4, the DuoRec model augmented with a sequence graph provides only minor improvements compared to our full LLMRG model. The DuoRec+GPT3.5 model without reasoning graph integration fails to significantly improve DuoRec performance on ML-1M, and even decreases performance on the Amazon Beauty dataset. Thanks to its greater capability, DuoRec+GPT4 boosts performance over DuoRec+GPT3.5 but still lags far behind our LLMRG model. These results demonstrate that the reasoning graph constructed by our proposed instructions is critical for performance, and simple next-item prediction is insufficient (DuoRec+GPT3.5 and DuoRec+GPT4). By explicitly modeling the reasoning process between user profiles and interaction sequences, LLMRG is able to make accurate, explainable recommendations. Our ablation studies confirm the reasoning graph's necessity and value in effectively leveraging the power of large language models for recommendation systems.

Our additional ablation studies further explore the effectiveness of each module in our LLMRG framework. Using DuoRec as a baseline model, we compared it to ablation versions of LLMRG with or without the divergent extension and self-verification modules based on GPT3.5 or GPT4. The results in Table 5 reveal that LLMRG (with GPT3.5 or GPT4) without the divergent extension module provides only marginal improvement compared with the complete LLMRG. However, removing the self-verification module from LLMRG (GPT3.5) actually decreases performance. This demonstrates the limited reasoning capability of GPT3.5 - without verification, uncontrolled reasoning introduces noise that reduces overall performance. Overall, these ablation experiments clearly demonstrate the value of both our divergent extension and self-verification modules in enabling more advanced reasoning while maintaining accuracy. The modules work synergistically to expand the search space of possible solutions while filtering out inaccurate or incoherent lines of reasoning.

Dataset	Metric	FDSA		BERT4Rec		CL4SRec		DuoRec	
		GPT3.5	GPT4	GPT3.5	GPT4	GPT3.5	GPT4	GPT3.5	GPT4
ML-1M	HR@5	+ 20.70%	+ 25.79%	+ 26.67%	+ 32.56%	+ 19.98%	+ 21.02%	+ 12.87%	+ 14.76%
	HR@10	+ 17.93 %	+ 22.87%	+ 13.52 %	+ 16.49 %	+ 17.30 %	+ 19.31 %	+ 14.10 %	+ 15.53 %
	ND@5	+ 21.33%	+ 30.27 %	+ 25.74 %	+ 32.82 %	+ 14.97 %	+ 16.78 %	+ 23.55 %	+ 26.01 %
	ND@10	+ 21.78%	+ 28.25%	+ 23.34%	+ 28.06%	+ 17.67%	+ 20.42%	+ 12.86%	+ 13.77%
Amazon Beauty	HR@5	+ 13.89 %	+ 17.53 %	+ 19.17 %	+ 23.22 %	+ 11.15 %	+ 14.15 %	+ 9.31 %	+ 11.93 %
	HR@10	+ 15.02 %	+ 17.78 %	+ 17.79 %	+ 22.14 %	+ 10.22 %	+ 11.32 %	+ 5.14 %	+ 6.61 %
	ND@5	+ 16.20 %	+ 18.64 %	+ 14.21 %	+ 17.63 %	+ 8.45 %	+ 10.18 %	+ 7.42 %	+ 9.24 %
	ND@10	+ 14.78 %	+ 17.64 %	+ 11.53 %	+ 14.76 %	+ 8.17 %	+ 9.68 %	+ 6.67 %	+ 7.95 %
Amazon Clothing	HR@5	+ 20.67 %	+ 23.92 %	+ 16.09 %	+ 19.10 %	+ 7.90 %	+ 10.92 %	+ 9.98 %	+ 11.40 %
	HR@10	+ 14.45 %	+ 17.88 %	+ 10.52 %	+ 13.72 %	+ 11.21 %	+ 14.99 %	+ 7.65 %	+ 9.48 %
	ND@5	+ 8.16 %	+ 10.86 %	+ 7.39 %	+ 10.39 %	+ 6.02 %	+ 9.09 %	+ 6.74 %	+ 9.19 %
	ND@10	+ 6.01 %	+ 8.13 %	+ 5.21 %	+ 5.94 %	+ 4.32 %	+ 8.07 %	+ 7.89 %	+ 9.29 %

Table 3: Performance comparison on three benchmark datasets. We set the original models as baselines (Table 2) to compare with our proposed LLMRG model based on GPT3.5 or GPT4. Higher is better.

Method	ML-1M				Amazon Beauty			
	HR@5	HR@10	ND@5	ND@10	HR@5	HR@10	ND@5	ND@10
DuoRec	0.2011	0.2837	0.1265	0.1663	0.0552	0.0839	0.0350	0.0447
DuoRec w/ seq	+ 6.36 %	+ 7.12 %	+ 12.25 %	+ 4.50 %	+ 3.26 %	+ 2.74 %	+ 3.71 %	+ 2.68 %
DuoRec+GPT3.5	+ 0.94 %	+ 0.81 %	+ 0.55 %	+ 1.80 %	**- 1.26 %**	**- 0.71 %**	**- 0.85 %**	**- 0.89 %**
LLMRG(GPT3.5)	+ 12.87 %	+ 14.10 %	+ 23.55 %	+ 12.86 %	+ 9.31 %	+ 5.14 %	+ 7.42 %	+ 6.67 %
DuoRec+GPT4	+ 3.28 %	+ 2.29 %	+ 3.95 %	+ 2.22 %	+ 0.72 %	+ 0.71 %	+ 0.86 %	+ 0.67 %
LLMRG(GPT4)	+ 14.76 %	+ 15.53 %	+ 26.01 %	+ 13.77 %	+ 11.93 %	+ 6.61 %	+ 9.24 %	+ 7.95 %

Table 4: Ablation studies of our LLMRG model on two benchmark datasets, i.e., ML-1M and Amazon Beauty. We take the DuoRec as a baseline model to compare with the DuoRec with sequence graph and DuoRec with direct recommendation results via naive GPT3.5 or GPT4 without constructing a reasoning graph. Higher is better.

LLM	Method	ML-1M				Amazon Beauty			
		HR@5	HR@10	ND@5	ND@10	HR@5	HR@10	ND@5	ND@10
NA	DuoRec	0.2011	0.2837	0.1265	0.1663	0.0552	0.0839	0.0350	0.0447
GPT3.5	w/o div	+ 5.12 %	+ 3.87 %	+ 8.30 %	+ 4.75 %	+ 3.62 %	+ 2.86 %	+ 4.57 %	+ 3.80 %
	w/o ver	**- 4.72 %**	**- 3.94 %**	**- 10.90 %**	**- 4.14 %**	**- 2.17 %**	**- 1.43 %**	**- 2.57 %**	**- 2.46 %**
	w/ div & ver	+ 12.87 %	+ 14.10 %	+ 23.55 %	+ 12.86 %	+ 9.31 %	+ 5.14 %	+ 7.42 %	+ 6.67 %
GPT4	w/o div	+ 7.06 %	+ 4.68 %	+ 13.35 %	+ 8.11 %	+ 4.89 %	+ 3.45 %	+ 4.28 %	+ 5.81 %
	w/o ver	+ 5.86 %	+ 2.36 %	+ 5.77 %	+ 3.72 %	+ 1.26 %	+ 1.31 %	+ 1.71 %	+ 1.56 %
	w/ div & ver	+ 14.76 %	+ 15.53 %	+ 26.01 %	+ 13.77 %	+ 11.93 %	+ 6.61 %	+ 9.24 %	+ 7.95 %

Table 5: Ablation studies of our LLMRG model on two benchmark datasets, i.e., ML-1M and Amazon Beauty. We take the DuoRec as a baseline model to compare with the ablation models w/ or w/o divergent extension and self-verification modules based on GPT3.5 or GPT4. Higher is better.

As shown in Figure 4, we also analyze the effectiveness of the proposed knowledge base self-improving. Based on LLMRG (GPT3.5), we calculate the average access frequency of the model call to LLM on two benchmark datasets. The experimental results show that the average access frequency decreases significantly as the number of reasoning steps increases. After 3,000 times of reasoning and verification, the average access frequency decreases by about 30% compared to not using this module, proving that the knowledge base contains high-quality reasoning chains that can be reused. Moreover, we observed that the reuse rate of high-quality reasoning chains in Amazon Beauty is higher than that of ML-1M, and the long-tailed distribution of Amazon products is one of the reasons for this difference.

To provide intuitive examples corroborating our quantitative results, we examined real case studies from the ML-1M dataset using (a) our complete LLMRG model, (b) LLMRG without the divergent extension module, and (c) LLMRG without the self-verification module. The case studies in Figure 2 illustrate the differences in reasoning be-

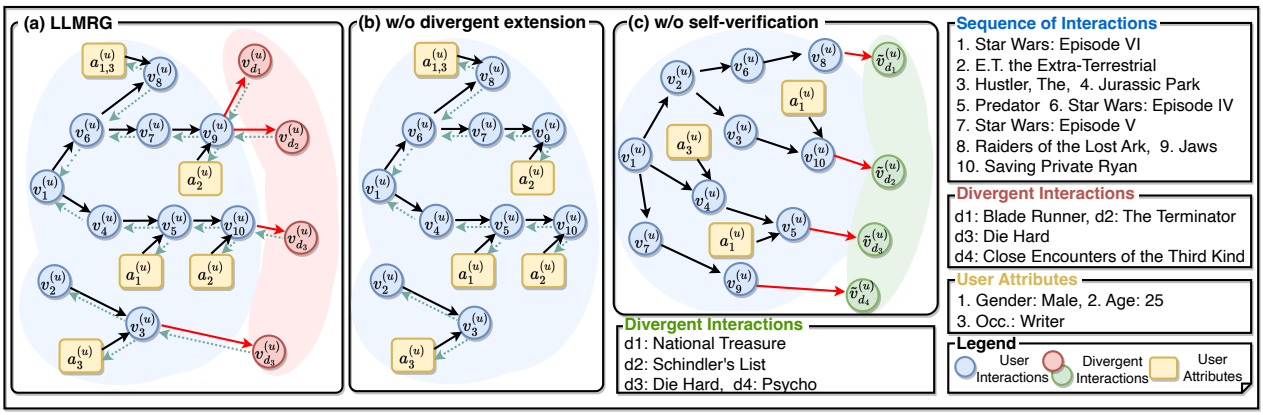

Figure 2: The real case studies (ML-1M) on our (a) LLMRG and ablation models, i.e., (b) LLMRG w/o divergent extension and (c) LLMRG w/o self-verification. The black arrow represents the reasoning procedure.

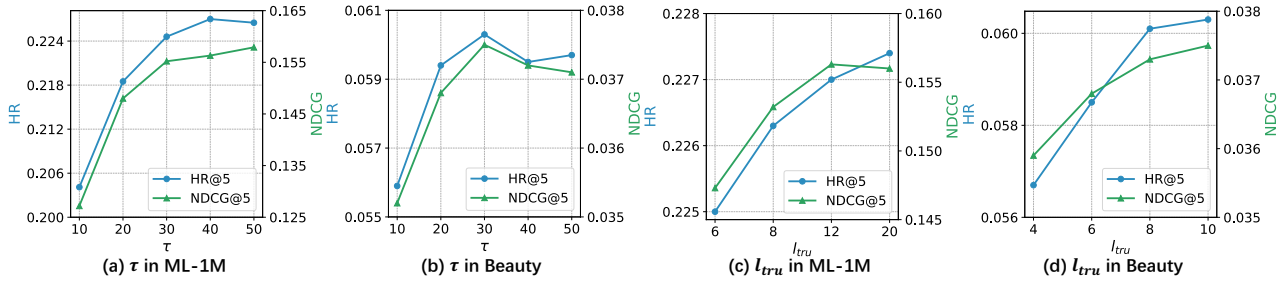

Figure 3: Sensitivity analysis of threshold of verification scoring τ and sequence truncation length l_{tru} on HR and NDCG performance based on ML-1M and Amazon Beauty benchmarks.

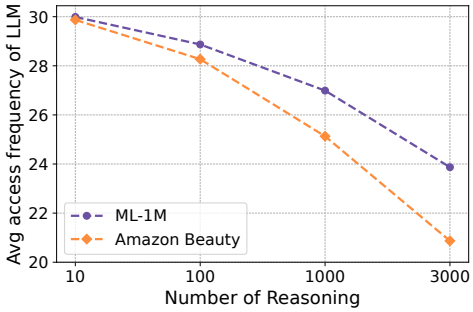

Figure 4: The average access frequency of LLM.

tween the models. LLMRG generates coherent recommendations with sound justifications, leveraging both divergent thinking to expand possibilities and self-verification to filter out poor options. Without divergent extensions, LLMRG struggles to move beyond obvious choices. Further, without self-verification, LLMRG's recommendations become more speculative and sometimes nonsensical, as the model lacks the ability to check its own thinking. These qualitative analyses mirror the patterns in our numerical results, serving as further validation of the value added by each reasoning module working in concert with our full LLMRG framework. The case studies provide intuitive examples of how our approach combines creative thinking and critical evaluation to produce logical recommendations.

Sensitivity Analysis. We evaluate LLMRG's sensitivity to the two most crucial parameters, τ and l_{tru}, on HR and NDCG, which control the threshold for verification scoring and sequence truncation length, respectively. Figure 3 (a) and (b) show that larger τ values yield more robust reasoning and filter out inferior options, thus boosting the model's performance on the ML-1M dataset. However, on the Beauty dataset, performance starts to decrease from $\tau = 30$, likely because higher verification scoring thresholds filter out more reasoning chains, increasing the sparsity of the graph. Figures 3 (c) and (d) indicate that, generally, longer sequences bring better recommendation results by incorporating more information. In summary, larger τ and longer sequences both tend to improve performance. τ exhibits a peak value, beyond which sparser reasoning graphs degrade results, especially for less logical sequences, such as Amazon products.

Conclusion

We present LLMRG that utilizes LLM to construct personalized reasoning graphs. This method demonstrates how LLM can bring logical reasoning and interpretability to recommendation systems without needing any additional information. We demonstrate that our plug-and-play method can effectively enhance multiple existing recommenders.

References

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chen, H.; Li, Y.; Sun, X.; Xu, G.; and Yin, H. 2021. Temporal meta-path guided explainable recommendation. In *Proceedings of the 14th ACM international conference on web search and data mining*, 1056–1064.

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Chu, Z.; Guo, H.; Zhou, X.; Wang, Y.; Yu, F.; Chen, H.; Xu, W.; Lu, X.; Cui, Q.; Li, L.; et al. 2023a. Data-Centric Financial Large Language Models. *arXiv preprint arXiv:2310.17784*.

Chu, Z.; Hao, H.; Ouyang, X.; Wang, S.; Wang, Y.; Shen, Y.; Gu, J.; Cui, Q.; Li, L.; Xue, S.; et al. 2023b. Leveraging large language models for pre-trained recommender systems. *arXiv preprint arXiv:2308.10837*.

Chu, Z.; Rathbun, S. L.; and Li, S. 2021. Graph infomax adversarial learning for treatment effect estimation with networked observational data. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 176–184.

Chu, Z.; Wang, Y.; Cui, Q.; Li, L.; Chen, W.; Li, S.; Qin, Z.; and Ren, K. 2024. LLM-Guided Multi-View Hypergraph Learning for Human-Centric Explainable Recommendation. *arXiv preprint arXiv:2401.08217*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Du, H.; Yuan, H.; Zhao, P.; Zhuang, F.; Liu, G.; Zhao, L.; Liu, Y.; and Sheng, V. S. 2023. Ensemble Modeling with Contrastive Knowledge Distillation for Sequential Recommendation. *arXiv preprint arXiv:2304.14668*.

Guan, Y.; Wang, D.; Chu, Z.; Wang, S.; Ni, F.; Song, R.; Li, L.; Gu, J.; and Zhuang, C. 2023. Intelligent Virtual Assistants with LLM-based Process Automation. *arXiv preprint arXiv:2312.06677*.

Harper, F. M.; and Konstan, J. A. 2015. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4): 1–19.

Li, S.; and Chu, Z. 2023. *Machine Learning for Causal Inference*. Springer Nature.

McAuley, J.; Targett, C.; Shi, Q.; and Van Den Hengel, A. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, 43–52.

Qiu, R.; Huang, Z.; Yin, H.; and Wang, Z. 2022. Contrastive learning for representation degeneration problem in sequential recommendation. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, 813–823.

Sheu, H.-S.; Chu, Z.; Qi, D.; and Li, S. 2021. Knowledge-guided article embedding refinement for session-based news recommendation. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12): 7921–7927.

Shin, R.; Lin, C. H.; Thomson, S.; Chen, C.; Roy, S.; Platanios, E. A.; Pauls, A.; Klein, D.; Eisner, J.; and Van Durme, B. 2021. Constrained language models yield few-shot semantic parsers. *arXiv preprint arXiv:2104.08768*.

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

Wang, H.; Zhang, F.; Zhang, M.; Leskovec, J.; Zhao, M.; Li, W.; and Wang, Z. 2019. Knowledge-aware graph neural networks with label smoothness regularization for recommender systems. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 968–977.

Wang, X.; Jin, H.; Zhang, A.; He, X.; Xu, T.; and Chua, T.-S. 2020. Disentangled graph collaborative filtering. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, 1001–1010.

Wang, Y.; Chu, Z.; Ouyang, X.; Wang, S.; Hao, H.; Shen, Y.; Gu, J.; Xue, S.; Zhang, J. Y.; Cui, Q.; et al. 2023. Enhancing recommender systems with large language model reasoning graphs. *arXiv preprint arXiv:2308.10835*.

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

Wu, S.; Tang, Y.; Zhu, Y.; Wang, L.; Xie, X.; and Tan, T. 2019. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 346–353.

Xie, X.; Sun, F.; Liu, Z.; Wu, S.; Gao, J.; Zhang, J.; Ding, B.; and Cui, B. 2022. Contrastive learning for sequential recommendation. In *2022 IEEE 38th international conference on data engineering (ICDE)*, 1259–1273. IEEE.

Xu, F.; Lin, Q.; Han, J.; Zhao, T.; Liu, J.; and Cambria, E. 2023. Are Large Language Models Really Good Logical Reasoners? A Comprehensive Evaluation From Deductive, Inductive and Abductive Views. *arXiv preprint arXiv:2306.09841*.

Xue, S.; Wang, Y.; Chu, Z.; Shi, X.; Jiang, C.; Hao, H.; Jiang, G.; Feng, X.; Zhang, J. Y.; and Zhou, J. 2023. Prompt-augmented temporal point process for streaming event sequence. *arXiv preprint arXiv:2310.04993*.

Zhang, T.; Zhao, P.; Liu, Y.; Sheng, V. S.; Xu, J.; Wang, D.; Liu, G.; Zhou, X.; et al. 2019. Feature-level Deeper Self-Attention Network for Sequential Recommendation. In *IJCAI*, 4320–4326.