Data-Centric Sequential Recommendation with Relation-Augmented Generation

Yichen Li[1,2], Yichen Tan[1], Yijing Shan[1], Haozhao Wang[1], Rui Zhang[1], Imran Razzak[2], Ruixuan Li[1*]

[1]School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China
[2]Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, United Arab Emirates
{ycli0204, hz_wang}@hust.edu.cn, (www.ruizhang.info)⟨rayteam@yeah.net⟩

Abstract

Data-Centric Sequential Recommendation (DaCSR) has emerged as a promising technique that enhances dataset quality to better capture user preferences without increasing training complexity. However, mining item relations to improve data quality remains challenging due to the intricate nature of interaction sequences. Existing methods predominantly either: 1) optimize models to learn such item relations from fixed datasets at significant training cost, or 2) employ generative models to adaptively learn only interaction patterns, which lack interpretability and cannot guarantee effective data quality enhancement. In this paper, we pioneer a relation-guided dataset augmentation and regeneration framework for sequential recommendation called **RaSR**. This framework can significantly improve model performance on original datasets while maintaining training efficiency without modifying the model architecture. Specifically, we first preprocess user interactions to construct standardized sequential data and extract semantic representations via a Large Language Model (LLM). We then build a multi-relation graph with manually predefined metrics and semantic representations to generate augmented datasets. Finally, a relation-aware generator can produce regenerated datasets with both the multi-relation graph and the augmented dataset. To verify the effectiveness of RaSR, we conduct experiments on various backbone models and datasets, and achieve significant performance improvement compared to training the model only on the original dataset.

Code — https://github.com/JY1230/RaSR

Introduction

Sequential recommendation has emerged as a critical research direction within recommender systems (Kang and McAuley 2018a; Sun et al. 2019a; Li et al. 2025a). Unlike conventional approaches that treat user interactions as independent events, this paradigm recognizes the inherently ordered temporal dependencies in behavioral sequences (Kang and McAuley 2018b; Sun et al. 2019b; Li et al. 2025b, 2024b). By modeling users' historical interaction trajectories, these systems capture the dynamics of user preferences to deliver more accurate recommendations.

*Ruixuan Li is the corresponding author.

Early research focused on item transition modeling via Markov chains (He and McAuley 2016; Chen et al. 2021; Liu et al. 2021). Recent advances leverage deep learning architectures, such as Transformers (Zhang et al. 2022; Chen, Pan, and Ming 2024), to better capture preference dynamics in sequential interactions (Tang and Wang 2018; Sun et al. 2019d; Shao et al. 2024; Li et al. 2020). These contributions collectively constitute a model-centric paradigm, aiming to optimize the backbone model within fixed datasets.

Although these methods prove effective, they inherently assume training data is high-quality and uncontaminated. In the era of big data, manually checking data quality can incur significant costs, and potential data flaws may substantially harm model convergence or necessitate substantial computational resources to detect and mitigate their negative impacts (Whang et al. 2023). Fortunately, data rectification has thus garnered widespread attention through systematic data screening and cleansing, which provides high-quality datasets that ensure model learning efficiency and performance (Li et al. 2021; Huang et al. 2024). (Guha et al. 2024) aims at fairness control by protecting sensitive attributes and optimizing Pareto-based bias mitigation. (Zhu et al. 2023) introduces multi-modal LLMs for visual-linguistic conflict detection and enhances cross-modal consistency. Within sequential recommendation, only (Yin et al. 2024) has addressed this issue and proposed a generative model named DR4SR to adaptively learn user interactions and regenerate training data for enhanced model performance.

In this paper, we investigate enhanced techniques for data rectification in sequential recommendation, referred to as Data-centric Sequential Recommendation (DaCSR). We first observe that DR4SR employs a sliding window to mine frequent subsets of user interactions, but fails to incorporate additional information to augment the data generation process. This limitation results in constrained generation quality and obscures interpretability due to the black-box nature of generative models, thereby hindering further optimization. In contrast, traditional Model-Centric Sequential Recommendation (MoCSR) research frequently utilizes item relations as additional information for model optimization, demonstrating substantial performance gains. How to effectively harness item relations for dataset rectification to generate higher-quality sequential data remains a fundamentally unexplored but pivotal research direction.

To tackle these challenges, we propose a novel Relation-augmented generation method for Sequential Recommendation named **RaSR** to achieve data rectification, thus enhancing model performance. More specifically, RaSR begins by preprocessing user interactions to construct standardized sequential data. We clean item metadata and filter some less frequent interactions. Concurrently, an LLM extracts semantic representations for items. We then build a multi-relation graph capturing three dependencies: 1) Co-occurrence relations from predefined metrics, 2) Temporal relations from predefined metrics, and 3) Semantic relations from LLM cosine similarities. This graph enables dual data augmentation: LLM-based strategies perform semantic modification, while explicit-relation strategies leverage predefined metrics for neighbor-aware modifications. These augmented sequences feed a relation-aware generator featuring a transformer encoder. The generator dynamically fuses co-occurrence features, temporal features, and projected LLM semantics via position-aware weighted fusion. Trained on source sequences, it regenerates rectified data to improve model performance.

To demonstrate the effectiveness of RaSR, we conduct experiments on six backbone models and four datasets to prove that RaSR can improve model performance compared to training the backbone model directly on the original dataset. As the sole existing baseline, DR4SR provides a critical comparison point. Our experiments demonstrate that RaSR regenerates a more effective dataset and delivers significantly greater model performance improvements. Our main contributions are summarized as follows:

- We pioneer a novel data-centric paradigm that addresses critical limitations of data quality in sequential recommendation. It overcomes the interpretability issues of existing data regeneration methods and the negative impacts of data flaws in model-centric methods.
- We propose a relation-guided augmentation-regeneration pipeline called RaSR. It constructs a multi-relation graph to generate augmented datasets, then employs a relation-aware transformer generator to produce regenerated sequences. This hybrid approach can produce interpretable and high-quality data rectification without requiring changes to the model architecture.
- Comprehensive validation across diverse settings demonstrates the effectiveness of RaSR. Experiments on six backbone models and four datasets show significant performance gains over original datasets, consistently outperforming the sole baseline.

Related Work

Sequential Recommendation. Sequential recommendation systems capture users' dynamic interests by modeling temporal dependencies and preference evolution in interaction histories (Fan et al. 2022; Li et al. 2023, 2025d,e, 2024a). Recent advances leverage attention mechanisms: SASRec (Kang and McAuley 2018c) uses transformer self-attention for long-term dependencies, while BERT4Rec (Sun et al. 2019c) employs bidirectional encoders for comprehensive modeling. To refine user and item representations, multi-source information integration has evolved. Early approaches incorporated multi-modal features like text and visuals (Ye et al. 2025). More recently, LLMs transform raw attributes into nuanced semantic representations, significantly boosting performance (Harte et al. 2023; Liu et al. 2024). However, existing methods predominantly focus on model-centric sequential recommendation that optimizes learning algorithms to enhance performance on fixed datasets. The authors in (Yin et al. 2024) initiated a paradigm shift toward data-centric approaches that address the impact of data quality. Following this research trajectory, we substantially extend these efforts to rectify data quality in sequential recommendation systems.

Relation-aware Recommendation. Unlike traditional methods relying solely on item-based collaborative filtering, relation-aware recommendation explicitly models item relations. One approach integrates items and attributes into knowledge graphs (KGs) (Wang et al. 2019; Yang et al. 2022; Wang et al. 2021), enhancing item representations via KG embedding tasks. CKE (Zhang et al. 2016) jointly optimizes KG embeddings and recommendations using item-attribute graphs. CFKG (Ai et al. 2018) incorporates users, defining a purchase relation to transform recommendations into KG completion. Another direction focuses on sequential scenarios, explicitly modeling relations between historical and target items (Wang et al. 2020b; Huang et al. 2019; Liu, Zhu, and Wu 2023). RCF (Xin et al. 2019) uses a two-level attention framework for relation types and intensities. KDA (Wang et al. 2020a) employs Fourier-based temporal evolution to model relational items over time. In addition to other works based on manually predefined metrics, the authors in (Yang et al. 2024) mine latent relations among items, enabling the model to adapt to diverse and complex recommendation tasks. However, existing methods primarily focus on leveraging item relations to enhance the training of recommendation models. This paper pioneers the use of item relations for dataset rectification by guiding generative models to synthesize higher-quality datasets.

Methodology

In this section, we first formulate the sequential recommendation process. Then, we introduced our relation-augmented generation method for sequential recommendation named RaSR. The workflow of RaSR is shown in Algorithm 1, and Fig.1 illustrates its framework.

Problem Formulation

Sequential Recommendation (SR) has emerged as a fundamental research domain in recommender systems, driven by user sequential behavioral data. Its core objective is to predict a user's next item of interest based on their ordered interaction history. The SR framework comprises three key stages: 1) Extract user behavioral sequences from interaction logs, representing them chronologically as item ID lists or embedding matrices; 2) Use the backbone model to capture dependencies within these sequences and learn latent feature representations of the patterns; 3) Leverage the modeled sequence features to predict the user's next item of inter-

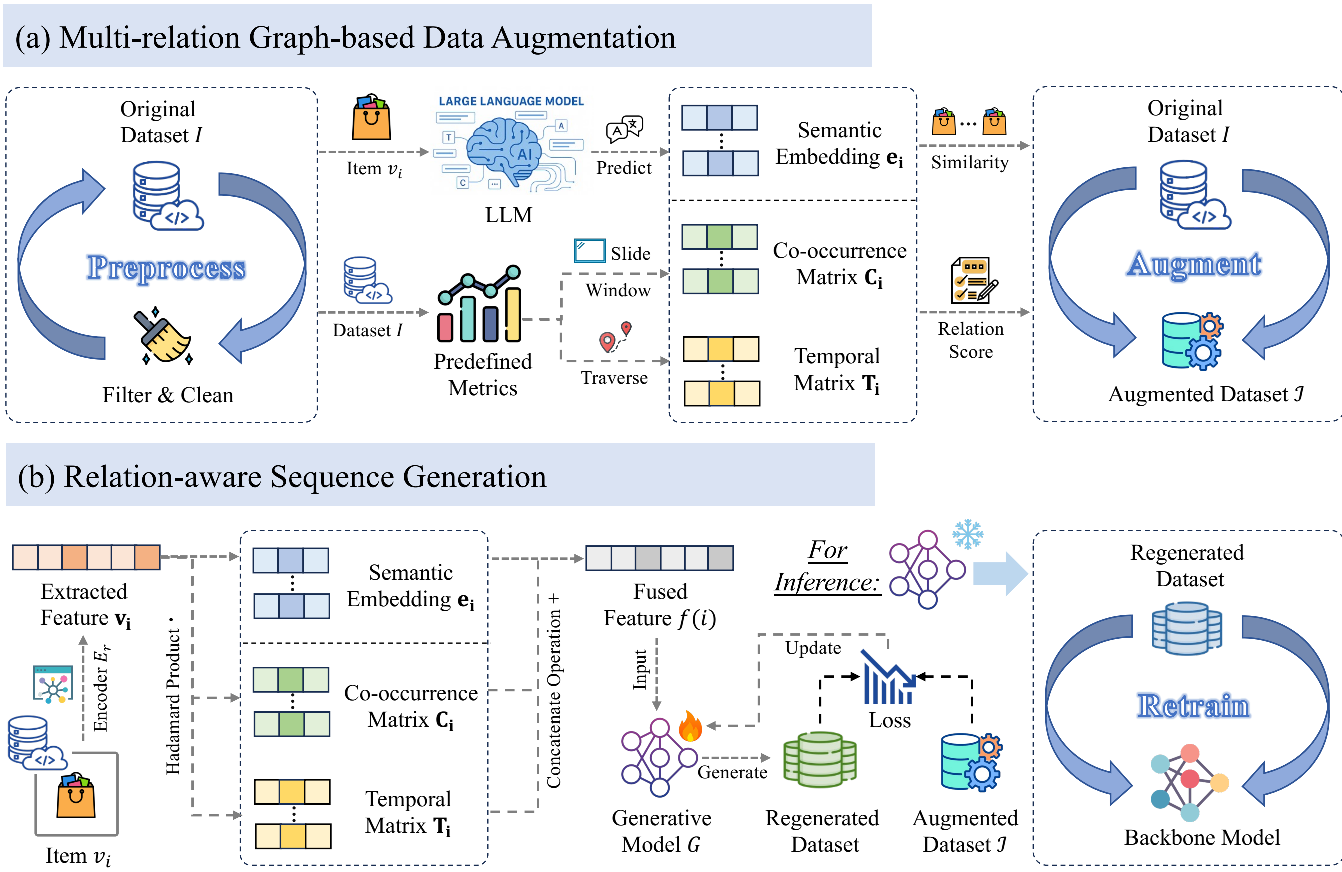

Figure 1: The framework of RaSR. RaSR enhances sequential recommendation data quality through a structured pipeline. It begins by preprocessing raw user interaction sequences and extracting semantic representations for items using the LLM. These embeddings, combined with predefined metrics, build a multi-relation graph capturing three item relations. This graph enables dual-path augmentation: LLM-based semantic modification and explicit-relation neighbor modification. The augmented data and graph then feed a relation-aware generator, which dynamically fuses the three relation types to regenerate the final dataset for training backbone models.

est based on current context or historical records, typically generating a Top-N recommendation list. This paradigm enhances recommender performance in dynamic environments by optimizing sequence modeling, delivering significant improvements to personalized services. Formally, given user set $\mathcal{U}$ and item set $\mathcal{V}$, for each user u with interaction sequence $I_u = [i_u^1, i_u^2, \ldots, i_u^{|I_u|}]$, the model learns parameters w by minimizing:

$$w^* = \arg\min_w \sum_{u \in \mathcal{U}} \sum_{t=1}^{|I_u|-1} \mathcal{L}\big(f_w(I_u^{1:t}), i_u^{t+1}\big). \tag{1}$$

where f_w encodes historical interactions for the backbone model w and $\mathcal{L}$ quantifies prediction loss function.

Framework and Workflow of RaSR

The key idea of RaSR is to leverage item relations to regenerate high-quality data for model training. We first construct standardized sequential data from user interactions and extract semantic representations using an LLM. Then, we build a graph with predefined metrics and LLM-derived semantics to generate augmented datasets. Finally, we produce regenerated datasets using both the multi-relational graph and augmented data via a relation-aware generator. We will illustrate the method with two components below: (1) Multi-relation Graph-based Data Augmentation, and (2) Relation-aware Sequence Regeneration.

Multi-relation Graph-based Data Augmentation. To ensure stable training of the regenerator, we first perform brief preprocessing on the original dataset. Raw interaction sequences are filtered to retain those with more than five interactions. Given the rich knowledge and semantic representation capabilities of the LLM model, we leverage the LLM to derive semantic embeddings for each item. For item v_i, its embedding $\mathbf{e}_i$ is computed as:

$$\mathbf{e}_i = \mathbf{W_1}(LLM(v_i)) + b_1. \tag{2}$$

where $LLM(\cdot)$ extracts the last-layer hidden output of the LLM to obtain item representations, and $\mathbf{W}$ and b denote the weight matrix and bias to align the dimension of representations and the backbone model.

Algorithm 1: RaSR

Input : $\mathcal{U}$: user set; $\mathcal{V}$: item set; I_u: original interaction for user u ; $\mathcal{I}_u$: augmented interaction for user u; w: backbone model; E_r: relation-aware conditional encoder; G: generative model.

Output: Regenerated Dataset.

Train the backbone model on the original dataset I with (1);

Multi-relation Graph-based Data Augmentation:
Preprocess the original dataset;
Obtain the item semantic embedding with (2);
Obtain the metrics-based relation $\{\mathbf{C}, \mathbf{T}\}$ with (3);
Compute the semantic-based relation **S** with (4);
Obtain the augmented dataset $\mathcal{I}$ with multi-relation graphs.

Relation-aware Sequence Regeneration:
Convert to item embedding $\mathbf{v}_i$ with relation-aware conditional encoder E_r;
Obtain the metric-based features $\{f_c(i), f_t(i)\}$ with (5);
Compute the semantic-based feature $f_s(i)$ with (6);
Fused features with position-aware gating $f(i)$ with (7);
Train the generative model G with fused features $f(i)$ and augmented dataset $\mathcal{I}$ with (8).

Obtain the regenerated dataset and enhance the backbone model with (1).

Then, we will build the multi-relation graph with the processed dataset and semantic embeddings. To better capture item relations within user interactions, we use both manually predefined metrics and the latent relations extracted by the LLM. The co-occurrence matrix scans the user sequence through a sliding window and calculates the co-occurrence frequency of products in the window, and the temporal matrix considers the adjacent positions of products in the sequence and calculates the time series dependency between adjacent products. These two metrics emphasize capturing temporal relationships between items defined as follows:

$$\mathbf{C}_{ij} = \sum_{S_u \in \mathcal{S}} \sum_{k=1}^{|S_u|-1} \mathbb{I}[(i_k, i_{k+1}) = (i, j)], \tag{3}$$

$$\mathbf{T}_{ij} = \sum_{S_u \in \mathcal{S}} \sum_{k=1}^{|S_u|-1} \lambda^k \cdot \mathbb{I}[i_k = i, i_{k+1} = j].$$

where $\mathbf{C}_{ij}$ denotes the co-occurrence matrix scanned via sliding window over sequences, and $\mathbf{T}_{ij}$ is the temporal matrix with decay factor λ. Moreover, we use the extracted semantic embeddings to build the semantic matrix with cosine similarity to capture the latent relations between items:

$$\mathbf{S}_{ij} = \frac{\mathbf{e}_i \cdot \mathbf{e}_j}{\|\mathbf{e}_i\| \|\mathbf{e}_j\|}. \tag{4}$$

Based on the multi-relation graph, we can obtain the augmented dataset with the following two strategies: 1)insert or substitute items with semantic similarity within user interaction sequences (i.e., at position t, insert item v_j with probability $\propto \mathbf{S}_{i_t,j}$); 2) leverage co-occurrence and temporal relation matrix to replace items guided by relation score (i.e., replace i_t v_j where $j = \arg\max_k(\mathbf{C}_{i_t,k} + \mathbf{T}_{i_t,k})$). We denote the augmented dataset as $(\mathcal{U}, \mathcal{V}, \mathcal{I} = [\mathbf{i}_u^1, \mathbf{i}_u^2, \dots, \mathbf{i}_u^{|\mathcal{I}_u|}])$, which enriches the semantic and pattern knowledge contributing to training the relation-aware sequence generator.

Relation-aware Sequence Regeneration. To enhance item relation in the regenerated dataset and thereby further improve data quality, we construct a relation-aware conditional encoder that performs position-aware weighted averaging of three multi-relation graph-based features. For each item v_i in the original dataset V, we first convert it to embedding $\mathbf{v}_i \in \mathbb{R}^d$ with the Transformer-based encoder. Then, we separately compute the co-occurrence and temporal features with $\mathcal{N}(i)$ neighbors for it:

$$f_c(i) = \sum_{j \in \mathcal{N}(i)} \mathbf{C}_{ij} \cdot \mathbf{v}_j, \quad f_t(i) = \sum_{j \in \mathcal{N}(i)} \mathbf{T}_{ij} \cdot \mathbf{v}_j. \tag{5}$$

For the semantic feature, we just apply a weight matrix $\mathbf{W_2}$ to project the LLM embedding:

$$f_s(i) = \mathbf{W}_s \mathbf{e}_i. \tag{6}$$

The fused features via position-aware gating can be formalized as:

$$f(i) = \alpha_i f_c(i) + \beta_i f_t(i) + \gamma_i f_s(i). \tag{7}$$

where $\alpha_i, \beta_i, \gamma_i$ are hyperparameters and we set 1 for each here. After obtaining relation-aware fused features through Eq.(7), any generative model (e.g., encoder-decoder architectures) can be employed for data regeneration. The RaSR framework and algorithm 1 demonstrate unique modularity, enabling direct integration of prior data generation algorithms as an off-the-shelf add-on. Unlike (Yin et al. 2024) where the authors project encoder features into K distinct vector spaces to capture diverse semantic information, our RaSR intrinsically integrates item relations within feature representations. Consequently, it imposes no additional constraints on the selection of generative models. Suppose the generation model is $G(\cdot) = (E, D)$, and the relation-aware conditional encoder is $E_r(\cdot)$, and the optimization process can be formulated as:

$$(G, E_r) = \arg\min_{(G, E_r)} \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \mathcal{L}(G, E_r; \mathcal{I}_u) \tag{8}$$

$$= -\frac{1}{|\mathcal{U}||\mathcal{I}_u|} \sum_{u \in \mathcal{U}} \sum_{t=1}^{|\mathcal{I}_u|} \log p_{(G,E_r)} \left(\mathbf{i}_u^t \mid \langle \text{sos} \rangle, i_u^1, \dots, i_u^{t-1} \right).$$

where $\langle$sos$\rangle$ denotes the start of the sequence. We employ the original dataset I as input and the augmented dataset $\mathcal{I}$ as regeneration targets to jointly train both the generative model G and relation-aware conditional encoder E_r, thereby ensuring information diversity in the regenerated dataset.

During the inference stage, to ensure diversity and stability in the regenerated dataset, we constrain the decoder's initial outputs: the first K items are preserved from the original

Dataset	Beauty				Sports				Toys				ML-1M			
Metrics	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20
GRU4Rec	0.0281	0.0492	0.0127	0.0180	0.0193	0.0319	0.0095	0.0126	0.0258	0.0424	0.0125	0.0167	0.1642	0.2632	0.0802	0.1051
Δ	15.30%↑	13.21%↑	16.54%↑	17.22%↑	18.13%↑	19.12%↑	17.89%↑	19.05%↑	17.83%↑	12.03%↑	14.40%↑	10.18%↑	5.79%↑	9.27%↑	4.24%↑	6.76%↑
DR4SR	0.0324	0.0557	0.0148	0.0211	0.0228	0.0380	0.0112	0.0150	0.0304	0.0475	0.0143	0.0184	0.1737	0.2876	0.0836	0.1122
Δ	18.86%↑	19.31%↑	20.47%↑	21.11%↑	20.73%↑	24.14%↑	22.11%↑	24.60%↑	21.32%↑	17.92%↑	21.60%↑	16.17%↑	13.22%↑	9.69%↑	14.84%↑	12.27%↑
RaSR (ours)	0.0334	0.0587	0.0153	0.0218	0.0233	0.0396	0.0116	0.0157	0.0313	0.0500	0.0152	0.0194	0.1859	0.2887	0.0921	0.1180
SASRec	0.0648	0.0966	0.0343	0.0423	0.0320	0.0512	0.0166	0.0214	0.0674	0.0955	0.0373	0.0444	0.1740	0.2737	0.0878	0.1129
Δ	5.71%↑	5.69%↑	8.16%↑	7.57%↑	12.19%↑	8.01%↑	16.87%↑	13.55%↑	16.62%↑	11.94%↑	20.91%↑	17.79%↑	8.39%↑	6.03%↑	9.68%↑	7.88%↑
DR4SR	0.0685	0.1021	0.0371	0.0455	0.0359	0.0553	0.0194	0.0243	0.0786	0.1069	0.0451	0.0523	0.1886	0.2902	0.0963	0.1218
Δ	7.41%↑	10.46%↑	8.75%↑	9.46%↑	24.38%↑	21.29%↑	29.52%↑	26.64%↑	20.03%↑	17.70%↑	24.40%↑	22.52%↑	11.21%↑	7.05%↑	12.87%↑	9.83%↑
RaSR (ours)	0.0696	0.1067	0.0373	0.0463	0.0398	0.0621	0.0215	0.0271	0.0809	0.1124	0.0464	0.0544	0.1935	0.2930	0.0991	0.1240
FMLP	0.0516	0.0826	0.0265	0.0343	0.0219	0.0364	0.0112	0.0149	0.0546	0.0806	0.0303	0.0369	0.1825	0.2947	0.0922	0.1205
Δ_{DR4SR}	9.11%↑	9.44%↑	5.28%↑	6.41%↑	17.81%↑	22.25%↑	13.39%↑	10.07%↑	18.68%↑	18.61%↑	9.90%↑	11.11%↑	4.88%↑	3.43%↑	3.58%↑	2.99%↑
DR4SR	0.0563	0.0904	0.0279	0.0365	0.0258	0.0445	0.0127	0.0164	0.0648	0.0956	0.0333	0.0410	0.1914	0.3048	0.0955	0.1241
Δ_{RaSR}	14.92%↑	18.16%↑	10.19%↑	13.12%↑	32.88%↑	36.26%↑	25.89%↑	24.16%↑	24.54%↑	12.90%↑	11.55%↑	18.16%↑	5.48%↑	5.90%↑	4.12%↑	4.65%↑
RaSR (ours)	0.0593	0.0976	0.0292	0.0388	0.0291	0.0496	0.0141	0.0185	0.0680	0.0910	0.0338	0.0436	0.1925	0.3121	0.0960	0.1261
GCE-GNN	0.0713	0.1033	0.0387	0.0468	0.0363	0.0575	0.0192	0.0245	0.0760	0.1048	0.0436	0.0509	0.1834	0.2795	0.0931	0.1171
Δ_{DR4SR}	16.55%↑	15.49%↑	11.89%↑	11.97%↑	21.76%↑	14.78%↑	25.00%↑	20.00%↑	15.66%↑	15.27%↑	15.37%↑	15.13%↑	1.36%↑	2.18%↑	0.64%↑	1.45%↑
DR4SR	0.0831	0.1193	0.0433	0.0524	0.0442	0.0660	0.0240	0.0294	0.0879	0.1208	0.0503	0.0586	0.1859	0.2856	0.0937	0.1188
Δ_{RaSR}	24.12%↑	19.94%↑	29.97%↑	26.50%↑	32.51%↑	25.39%↑	34.90%↑	30.20%↑	17.63%↑	17.65%↑	16.28%↑	16.31%↑	8.07%↑	8.62%↑	7.84%↑	8.37%↑
RaSR (ours)	0.0885	0.1239	0.0503	0.0592	0.0481	0.0721	0.0259	0.0319	0.0894	0.1233	0.0507	0.0592	0.1982	0.3036	0.1004	0.1269
CL4Rec	0.0668	0.1004	0.0359	0.0444	0.0348	0.0531	0.0187	0.0233	0.0704	0.1026	0.0392	0.0473	0.1546	0.2392	0.0757	0.0970
Δ_{DR4SR}	8.08%↑	7.57%↑	7.80%↑	5.86%↑	4.89%↑	6.21%↑	4.81%↑	6.01%↑	4.97%↑	4.39%↑	3.32%↑	3.59%↑	2.39%↑	1.21%↑	1.06%↑	1.55%↑
DR4SR	0.0722	0.1080	0.0387	0.0470	0.0365	0.0564	0.0196	0.0247	0.0739	0.1071	0.0405	0.0490	0.1583	0.2421	0.0765	0.0985
Δ_{RaSR}	8.53%↑	8.96%↑	10.86%↑	7.43%↑	8.62%↑	7.72%↑	8.02%↑	8.58%↑	6.53%↑	5.75%↑	5.61%↑	3.38%↑	3.10%↑	2.80%↑	2.91%↑	3.51%↑
RaSR (ours)	0.0725	0.1094	0.0398	0.0477	0.0378	0.0572	0.0202	0.0253	0.0750	0.1085	0.0414	0.0489	0.1594	0.2459	0.0779	0.1004

Table 1: Performance comparison of different baselines with various backbone models. Δ denotes the performance improvement compared. The best results are bold.

sequence. This maintains core sequential integrity while anchoring the generation process. Subsequent items are generated autonomously through the decoder prediction until the generation is completed. Finally, we will enhance the backbone model with the regenerated dataset followed as Eq.(1).

Experiments

We conduct experiments on four datasets using various backbone models. We record the test accuracy and performance improvement to evaluate RaSR. Moreover, we provide an ablation study and parameter sensitivity analysis to explore each module in RaSR.

Setup

Datasets. To validate our approach, we conducted experiments similar to those in (Yin et al. 2024; Li et al. 2025c) on four open-source datasets: three subsets (**Beauty, Sports, Toys**) from Amazon review data (Ni, Li, and McAuley 2019), and **MovieLens-1M** (ML-1M) (Harper and Konstan 2015). We list more details about these datasets in Table 2.

Datasets	Beauty	Sports	Toys	ML-1M
User Number	22,363	35,598	19,412	6,040
Item Number	12,101	18,357	11,924	3,952
Interaction	198,502	296,337	167,597	1,000,209
Average Length	8.88	8.32	8.63	165.60
Sparsity	99.93%	99.95%	99.93%	95.81%

Table 2: Details of four datasets.

Baselines. For a fair comparison, we only select DR4SR (Yin et al. 2024) as the baseline because RaSR aims to rectify the dataset via regenerating a high-quality dataset, and this research topic still remains fundamentally unexplored. Considering that both RaSR and DR4SR can be seen as off-the-shelf add-ons to enhance other model-centric methods, we select five classic backbone models to demonstrate that RaSR can effectively enhance model performance with the regenerated dataset: GRU4Rec (Hidasi et al. 2015), SASRec (Kang and McAuley 2018d), FMLP (Zhou et al. 2022), GCE-GNN (Wang et al. 2020c), CL4Rec (Xie et al. 2022). Also, these five backbone models have been used in (Yin et al. 2024) to verify the method's effectiveness.

Configurations. Unless otherwise mentioned, We use Adam as an optimizer with a linear learning rate schedule, and the learning rate is set to 1e-3, with a batch size of 256 and embedding dimension of 64. For all datasets, the maximum length of historical sequences is set to 50. During the data augmentation phase, the maximum number of variants for each interaction sequence is set to 4, with 2 variants derived from semantic similarity and 2 from relation scores, using a window size of 2 and a decay factor of 0.8. When training the generative model, the number of training epochs is uniformly set to 100, and the diversity factor is set to 5. The best checkpoints are selected to generate the regenerated dataset. All experiments are run on 8 RTX 4090 GPUs and 16 RTX 3090 GPUs. We set the remaining parameters according to the values in the original open-source code.

Performance Overview

Test Accuracy. Across all four datasets and five distinct backbone architectures in Table 1, RaSR consistently elevates test-accuracy metrics compared to both the original training regime and the prior data-regeneration baseline. Notably, the gains are uniform across model families, demonstrating that enriching the training data via relation-guided augmentation and regeneration is model-agnostic and broadly effective. This universality suggests that RaSR successfully addresses fundamental data sparsity and long-tail challenges inherent in sequential recommendation. By injecting semantically and structurally informed variations into the training sequences, RaSR enables each model to learn richer patterns without altering its architecture or capacity. Moreover, the impact of RaSR is particularly pronounced in sparser interaction contexts, where conventional backbones struggle to capture rare but informative user–item transitions. In these settings, the relation-aware generator fills gaps in the data distribution by synthesizing plausible yet diverse behaviors, thereby improving the robustness of learned representations. RaSR amplifies key relational information in low-frequency scenarios and sharpens predictive accuracy, making it a compelling add-on for any sequential recommendation system.

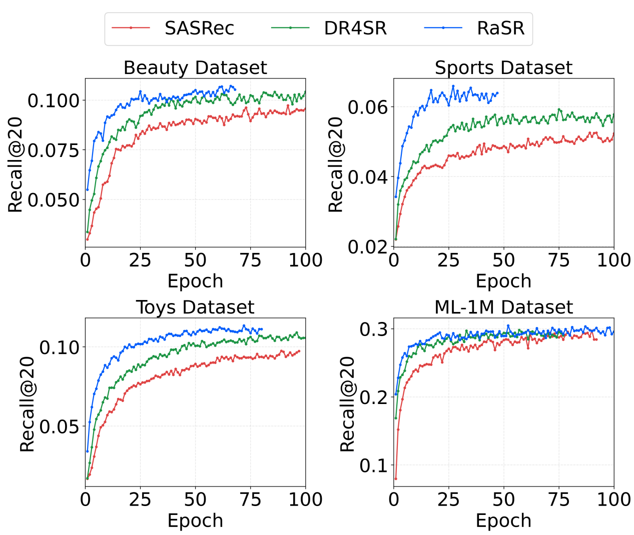

Figure 2: The convergence trend of RaSR and other baselines. We report performance at each round and apply an early stopping strategy here: training will stop if N@20 on the validation set fails to improve for 20 rounds. Both original and regenerated data are used for DR4SR and RaSR.

Convergence Trend. Fig.2 presents the convergence trend of RaSR compared to both DR4SR and the original training regime across multiple training rounds. A consistent and notable observation is that RaSR exhibits a faster and more stable convergence trajectory under the early stopping strategy, which terminates training when NDCG@20 on the validation set plateaus for 20 consecutive rounds. Specifically, RaSR not only reaches a higher performance plateau within fewer iterations but also maintains a smoother ascent curve throughout the optimization process. This indicates that the relation-guided augmentation and regeneration mechanisms incorporated in RaSR significantly enhance the quality of training data, enabling the backbone models to extract more discriminative sequence patterns at an early stage. In contrast, DR4SR, despite being a data-centric method as well, shows less stable convergence with evident fluctuations, likely due to its weaker utilization of item relations and less structured generation pipeline. The original training regime, which relies solely on raw sequences, converges more slowly and suffers from inferior generalization, as reflected in its lower Recall scores. These trends highlight that by injecting relational inductive bias through multi-relation graphs and semantic fusion, RaSR improves data efficiency and reduces the number of redundant or misleading samples that often hinder learning in sparse sequential domains. Importantly, this convergence advantage is achieved without modifying the model architecture or introducing additional inference overhead, making RaSR a practical and effective plug-in for diverse backbone models.

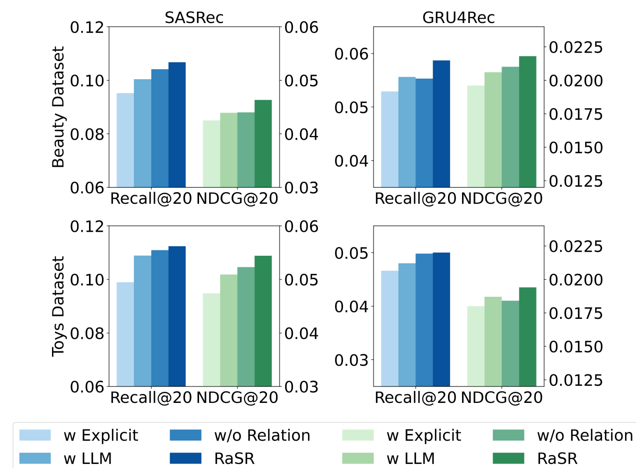

Figure 3: The ablation study of three components in RaSR.

Ablation Study. To investigate the individual contributions of each component in the RaSR framework, we conduct an ablation study, as summarized in Fig.3. Specifically, we analyze three core modules: (1) the semantic-aware augmentation based on LLM-derived item embeddings, (2) the explicit relation-aware augmentation via co-occurrence and temporal matrices, and (3) the relation-aware sequence regeneration module, which fuses multi-relation features for generative modeling. By selectively removing each module and observing the impact on overall performance, we aim to evaluate their respective effectiveness in improving recommendation accuracy.

The results reveal that all three components can contribute to the performance gains of RaSR, but to different extents. Removing the relation-aware sequence generator leads to the most significant performance degradation. This highlights the importance of augmenting the data and effectively encoding and generating sequences that preserve and exploit complex relational structures. Without this mod-

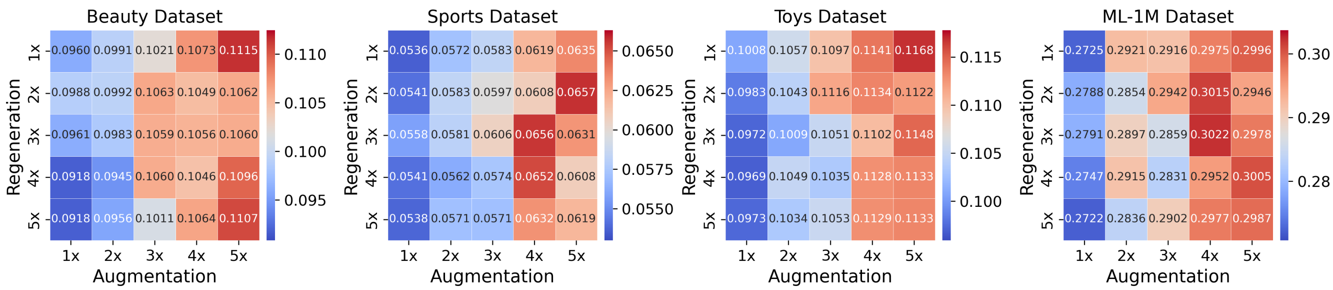

Figure 4: The analysis of the number of regenerated data in RaSR. We perform a grid search by adjusting the ratio of augmented and regenerated data relative to the original data on all datasets.

ule, the model cannot integrate the relation-enhanced representations into the regeneration process, resulting in lower-quality synthetic data. Eliminating the semantic augmentation component also results in a noticeable drop in performance. While this module alone may not provide as much improvement as the generator, it plays a crucial role in introducing latent, non-local similarities that are otherwise inaccessible to pure interaction-based metrics. This is especially important in sparse datasets, where co-occurrence information is weak and the LLM-derived semantic relations offer a meaningful supplement to capture latent user intent. Lastly, the explicit relation augmentation proves to be a strong complement to semantic modeling. Its removal causes a moderate decline, suggesting that predefined statistical patterns still provide essential inductive bias.

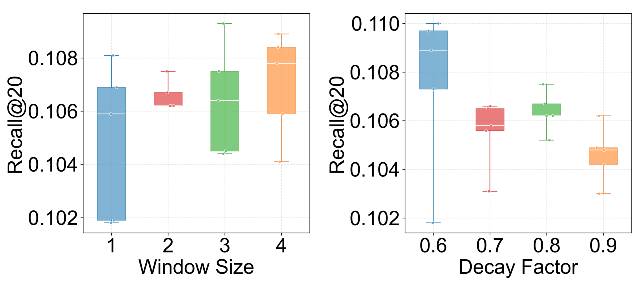

Figure 5: Performance of RaSR under different configurations (a) window size in the co-occurrence matrix, (b) decay factor in the temporal matrix on Beauty with SASRec.

Parameter Sensitivity Analysis. To further investigate the impact of data generation quantity, we conducted a comprehensive parameter sensitivity analysis by varying the ratio of augmented and regenerated data relative to the original dataset. Specifically, we performed a grid search on all four datasets to evaluate how different proportions affect model performance. As shown in Fig.4, increasing the generated data generally leads to performance improvement at the early stages. However, the gains tend to saturate or even slightly degrade after a certain threshold. This suggests that while relation-augmented data generation helps mitigate issues such as sparsity and noise in the original data, excessive synthetic data may introduce distributional shifts that impair the model's ability to learn from authentic behavior patterns. Therefore, controlling the generation ratio is crucial to balance informational diversity and distributional fidelity.

Moreover, the sensitivity to the data generation ratio varies across datasets. For sparser ones such as Beauty and Toys, increasing synthetic data substantially improves the model's ability to capture long-tail preferences. In denser datasets like ML-1M, however, the gains from additional generated sequences are less pronounced, likely because they already contain abundant interaction signals. These observations suggest that the generation ratio should be adaptively chosen according to dataset characteristics. We also examine two key hyperparameters in the multi-relation graph: the window size in the co-occurrence matrix and the decay factor in the temporal matrix. As shown in Fig.5, a moderately larger window size enhances performance by capturing broader item co-occurrences, though excessively large windows may incorporate noise. Similarly, an appropriate decay factor balances recent interactions with long-term patterns, achieving optimal results. Both findings underscore the need to carefully calibrate relation extraction parameters for meaningful and effective data augmentation.

Conclusion

We introduce RaSR, a novel relation-augmented generation framework for data-centric sequential recommendation. RaSR pioneers a multi-relation graph construction that captures manually predefined metrics and semantic dependencies for dataset augmentation. Its relation-aware generator dynamically fuses these features to regenerate a high-quality dataset, enhancing backbone models without architectural changes. Extensive experiments have demonstrated that RaSR can empirically improve recommendation performance across various settings.

Acknowledgments

This work is supported by the National Key Research and Development Program of China under grant 2024YFC3307900; the National Natural Science Foundation of China under grants 625B2073, 62376103, 62302184 and 62436003; Major Science and Technology Project of Hubei Province under grants 2025BAB011 and 2024BAA008; and Hubei Science and Technology Talent Service Project under grant 2024DJC078.

References

Ai, Q.; Azizi, V.; Chen, X.; and Zhang, Y. 2018. Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms*, 11(9): 137.

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.

Chen, M.; Pan, W.; and Ming, Z. 2024. Explicit and implicit modeling via dual-path transformer for behavior set-informed sequential recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 329–340.

Fan, Z.; Liu, Z.; Wang, Y.; Wang, A.; Nazari, Z.; Zheng, L.; Peng, H.; and Yu, P. S. 2022. Sequential recommendation via stochastic self-attention. In *Proceedings of the ACM web conference 2022*, 2036–2047.

Guha, S.; Khan, F. A.; Stoyanovich, J.; and Schelter, S. 2024. Automated data cleaning can hurt fairness in machine learning-based decision making. *IEEE Transactions on Knowledge and Data Engineering*, 36(12): 7368–7379.

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.

Harte, J.; Zorgdrager, W.; Louridas, P.; Katsifodimos, A.; Jannach, D.; and Fragkoulis, M. 2023. Leveraging large language models for sequential recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, 1096–1102.

He, R.; and McAuley, J. 2016. Fusing similarity models with markov chains for sparse sequential recommendation. In *2016 IEEE 16th international conference on data mining (ICDM)*, 191–200. IEEE.

Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; and Tikk, D. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*.

Huang, W.; Liu, Y.; Ye, M.; Chen, J.; and Du, B. 2024. Federated learning with long-tailed data via representation unification and classifier rectification. *IEEE Transactions on Information Forensics and Security*, 19: 5738–5750.

Huang, X.; Fang, Q.; Qian, S.; Sang, J.; Li, Y.; and Xu, C. 2019. Explainable interaction-driven user modeling over knowledge graph for sequential recommendation. In *proceedings of the 27th ACM international conference on multimedia*, 548–556.

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

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

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

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

Li, J.; Wang, M.; Li, J.; Fu, J.; Shen, X.; Shang, J.; and McAuley, J. 2023. Text is all you need: Learning language representations for sequential recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1258–1267.

Li, M.; Zareian, A.; Lin, Y.; Pan, X.; Whitehead, S.; Chen, B.; Wu, B.; Ji, H.; Chang, S.-F.; Voss, C.; et al. 2020. GAIA: A fine-grained multimedia knowledge extraction system. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 77–86.

Li, P.; Rao, X.; Blase, J.; Zhang, Y.; Chu, X.; and Zhang, C. 2021. Cleanml: A study for evaluating the impact of data cleaning on ml classification tasks. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, 13–24. IEEE.

Li, Y.; Li, Q.; Wang, H.; Li, R.; Zhong, W.; and Zhang, G. 2024a. Towards Efficient Replay in Federated Incremental Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12820–12829.

Li, Y.; Qin, Q.; Zhu, G.; Xu, W.; Wang, H.; Li, Y.; Zhang, R.; and Li, R. 2025a. A Systematic Survey on Federated Sequential Recommendation. arXiv:2504.05313.

Li, Y.; Shan, Y.; LIU, Y.; Wang, H.; Wang, C.; wangshi.ww; Wang, Y.; and Li, R. 2025b. Efficient Knowledge Transfer in Federated Recommendation for Joint Venture Ecosystem. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Li, Y.; Shan, Y.; Liu, Y.; Wang, H.; Wang, W.; Wang, Y.; and Li, R. 2025c. Personalized Federated Recommendation for Cold-Start Users via Adaptive Knowledge Fusion. In *Proceedings of the ACM on Web Conference 2025*, 2700–2709.

Li, Y.; Wang, H.; Qi, Y.; Liu, W.; and Li, R. 2025d. Re-fed+: A better replay strategy for federated incremental learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Li, Y.; Wang, H.; Xu, W.; Xiao, T.; Liu, H.; Tu, M.; Wang, Y.; Yang, X.; Zhang, R.; Yu, S.; Guo, S.; and Li, R. 2024b. Unleashing the Power of Continual Learning on Non-Centralized Devices: A Survey. arXiv:2412.13840.

Li, Y.; Wang, Y.; Wang, H.; Qi, Y.; Xiao, T.; and Li, R. 2025e. FedSSI: Rehearsal-Free Continual Federated Learning with Synergistic Synaptic Intelligence. In *Forty-second International Conference on Machine Learning*.

Liu, C.; Li, X.; Cai, G.; Dong, Z.; Zhu, H.; and Shang, L. 2021. Noninvasive self-attention for side information fusion in sequential recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 4249–4256.

Liu, H.; Zhu, Y.; and Wu, Z. 2023. Knowledge graph-based behavior denoising and preference learning for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 36(6): 2490–2503.

Liu, Q.; Wu, X.; Wang, Y.; Zhang, Z.; Tian, F.; Zheng, Y.; and Zhao, X. 2024. Llm-esr: Large language models enhancement for long-tailed sequential recommendation. *Advances in Neural Information Processing Systems*, 37: 26701–26727.

Ni, J.; Li, J.; and McAuley, J. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, 188–197.

Shao, M.; Huang, H.; Peng, Q.; and Liu, H. 2024. ULMRec: User-centric large language model for sequential recommendation. *arXiv preprint arXiv:2412.05543*.

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

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

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

Sun, F.-Y.; Hoffmann, J.; Verma, V.; and Tang, J. 2019d. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. *arXiv preprint arXiv:1908.01000*.

Tang, J.; and Wang, K. 2018. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the eleventh ACM international conference on web search and data mining*, 565–573.

Wang, C.; Ma, W.; Zhang, M.; Chen, C.; Liu, Y.; and Ma, S. 2020a. Toward dynamic user intention: Temporal evolutionary effects of item relations in sequential recommendation. *ACM Transactions on Information Systems (TOIS)*, 39(2): 1–33.

Wang, P.; Fan, Y.; Xia, L.; Zhao, W. X.; Niu, S.; and Huang, J. 2020b. KERL: A knowledge-guided reinforcement learning model for sequential recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 209–218.

Wang, X.; He, X.; Cao, Y.; Liu, M.; and Chua, T.-S. 2019. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 950–958.

Wang, X.; Huang, T.; Wang, D.; Yuan, Y.; Liu, Z.; He, X.; and Chua, T.-S. 2021. Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the web conference 2021*, 878–887.

Wang, Z.; Wei, W.; Cong, G.; Li, X.-L.; Mao, X.-L.; and Qiu, M. 2020c. Global context enhanced graph neural networks for session-based recommendation. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, 169–178.

Whang, S. E.; Roh, Y.; Song, H.; and Lee, J.-G. 2023. Data collection and quality challenges in deep learning: A data-centric ai perspective. *The VLDB Journal*, 32(4): 791–813.

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.

Xin, X.; He, X.; Zhang, Y.; Zhang, Y.; and Jose, J. 2019. Relational collaborative filtering: Modeling multiple item relations for recommendation. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, 125–134.

Yang, S.; Ma, W.; Sun, P.; Ai, Q.; Liu, Y.; Cai, M.; and Zhang, M. 2024. Sequential recommendation with latent relations based on large language model. In *Proceedings of the 47th International ACM SIGIR conference on research and development in information retrieval*, 335–344.

Yang, Y.; Huang, C.; Xia, L.; and Li, C. 2022. Knowledge graph contrastive learning for recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, 1434–1443.

Ye, Y.; Zheng, Z.; Shen, Y.; Wang, T.; Zhang, H.; Zhu, P.; Yu, R.; Zhang, K.; and Xiong, H. 2025. Harnessing multimodal large language models for multimodal sequential recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 13069–13077.

Yin, M.; Wang, H.; Guo, W.; Liu, Y.; Zhang, S.; Zhao, S.; Lian, D.; and Chen, E. 2024. Dataset Regeneration for Sequential Recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24: The 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery. ISBN 979-84-00-70490-1.

Zhang, F.; Yuan, N. J.; Lian, D.; Xie, X.; and Ma, W.-Y. 2016. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 353–362.

Zhang, M.; Wu, S.; Yu, X.; Liu, Q.; and Wang, L. 2022. Dynamic graph neural networks for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(5): 4741–4753.

Zhou, K.; Yu, H.; Zhao, W. X.; and Wen, J.-R. 2022. Filter-enhanced MLP is all you need for sequential recommendation. In *Proceedings of the ACM web conference 2022*, 2388–2399.

Zhu, Z.; Zhang, M.; Wei, S.; Wu, B.; and Wu, B. 2023. Vdc: Versatile data cleanser based on visual-linguistic inconsistency by multimodal large language models. *arXiv preprint arXiv:2309.16211*.