Diffusion Based Data Augmentation for Multi-behavior Sequential Recommendation

Tingting Zheng[1], Zhilong Shan[1](✉), Zhengyang Wu[1], and Xiaoyong HU[2]

[1] School of Computer Science, South China Normal University, Guangzhou 510631, People's Republic of China
{2019010180,zlshan}@m.scnu.edu.cn
[2] Institute of Artificial Intelligence in Education, South China Normal University, Guangzhou 510631, People's Republic of China

Abstract. Recommender systems play a pivotal role in tackling information overload but are often faced with problems of data sparsity and cold start. Multi-Behavior Sequential Recommendation (MBSR) tackles these challenges by taking diverse user-item interaction behaviors into account. However, existing MBSR models are often challenged by the imbalance issue in multi-behavior data. Traditional data augmentation techniques for MBSR are insufficient as they concentrate on single-behavior data and fail to account for the interplay between behaviors and the dynamics of sequences. To address these shortcomings, we propose a diffusion based data augmentation framework for multi-behavior sequential recommendation (DiffAMB). We utilize diffusion models and contrastive learning to generate high quality data for MBSR models. This framework enables the augmentation of multiple-behavior datasets while preserving the logical consistency of user sequences. Our experiments on real-world datasets confirm the effectiveness of our approach.

Keywords: Recommender systems · Multi-Behavior Sequential Recommendation · Diffusion model · Contrastive learning.

1 Introduction

Recommender systems are widely used in e-commerce, news, and social media but face challenges like data sparsity, cold start, and long-tail users, leading to poor recommendations for infrequent users. Multi-Behavior Recommendation (MBR) addresses these issues by leveraging diverse user-item interactions to improve performance. Multi-Behavior Sequential Recommendation (MBSR) further utilizes dynamic collaborative signals from users' behavioral sequences. With advancements in deep learning, MBSR models increasingly employ Graph Neural Networks (GNNs), their variants[1], and hyper-graphs[2], often integrating attention mechanisms to enhance representation learning and better capture user behavior characteristics and relations[4,5].

Despite advancements in MBSR models, challenges like data imbalance and sparsity persist, particularly with target behaviors (e.g., purchases) being sparser than auxiliary behaviors (e.g., clicks). To address this, data augmentation methods, including contrastive learning[3], sequence-level[6], and item-level strategies [7,8], have been proposed. However, not all augmentations are beneficial[3], and limitations remain: (1) Most methods focus on single-behavior recommendations and are not directly applicable to MBSR. (2) Few studies consider behavioral relations. (3) Sequence-level strategies may fail for users with short interaction sequences, while item-level strategies are less explored[6]–[8].

To address these challenges, we propose DiffAMB, a diffusion-based data augmentation framework for MBSR. Unlike prior single-behavior augmentation methods, DiffAMB focuses on capturing the correlations among diverse behaviors and incorporating the sequential dynamics of user interactions. This ensures the augmented data are both authentic and logically coherent. In summary, our main contributions of this paper are as follows:

- We propose DiffAMB, a data augmentation framework that generates high-quality, logically consistent data for MBSR datasets, compatible with existing MBSR models utilizing multiple auxiliary behaviors.
- We enhance the diffusion model and integrate contrastive learning to produce diverse behavior sequences, effectively capturing user preferences and inter-behavior relationships to improve data augmentation.
- DiffAMB offers a novel approach to tackling data sparsity and cold-start issues in MBSR, demonstrating the potential of diffusion models to enhance recommendations for users with sparse interaction histories.
- We validate DiffAMB's effectiveness through extensive experiments and detailed analysis on real-world datasets.

2 Related Work

Data augmentation in recommender systems mitigates data sparsity by generating user sequences or pseudo-interaction items[6]–[7]. However, most methods focus on single-behavior recommendation, with MBASR[3] being an exception, though limited to one auxiliary behavior and unsuitable for multiple auxiliary behaviors. Contrastive learning (CL), a self-supervised method, improves data representation by maximizing similarity for positive pairs and minimizing it for negative pairs. In sequential recommendation, CL applies transformations (e.g., rotation, cropping) to generate augmented views[11], resembling graph-based methods[1]. For multi-behavior sequential recommendation (MBSR), multi-view interactions help identify shared and unique behavioral traits[12]. However, integrating these views into a cohesive framework while ensuring augmented data reflects real-world behavior remains challenging.

3 Problem Formulation

In MBSR, let U and V represent the user and item sets, with sizes $|U|$ and $|V|$, respectively. The set of interaction behaviors between users and items is denoted as B, with $|B| = K$ behavior types. Here, the K-th behavior is defined as the target behavior, while the remaining $K-1$ behaviors are auxiliary. For a user $u_i \in U$, the interaction sequence can be written as $S_i = \{(v_{i,1}, b_{i,1}), (v_{i,2}, b_{i,2}), ..., (v_{i,M}, b_{i,M})\}$, $v_{i,j} \in V, b_{i,j} \in B$. M is the length of the interaction sequence which is also denoted as the maximum time step. Additionally, the subsequence of S_i with the same specific behavior b_k can be denoted as $S_{i,k}$. The task of MBSR is defined as follows.

Definition 1. *Task of MBSR. Given a user $u_i \in U$ with the interaction sequence S_i of length M, where $S_i = \{(v_{i,1}, b_{i,1}), (v_{i,2}, b_{i,2}), ..., (v_{i,M}, b_{i,M})\}$. $v_{i,j} \in V$ and $b_{i,j} \in B$. The task of the recommendation is to predict the probability that user u_i interacts with item $v_{i,M+1} \in V$ under the target behavior b_K at timestep $M+1$.*

4 Methodology

4.1 Overall Framework

In this section, we will detail our diffusion-based data augmentation framework(DiffAMB) as shown in Fig. 1 . We adopt a behavior-aware user sequence representation, integrating item embeddings, positional embeddings, and behavior type embeddings. Diffusion models are used for MBSR data augmentation. During diffusion, two types of contrastive learning control noise generation. Discrete augmented sequences are then refined through a behavioral logic check and completion step to preserve behavior dependencies.

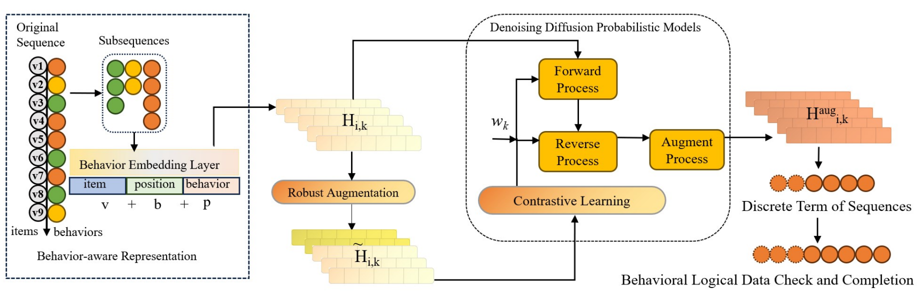

Fig. 1. DiffAMB overall framework. DiffAMB generally involves interaction sequences embedding, denoising diffusion probabilistic models with contrastive learning, behavioral logical checks and data completion.

4.2 Behavior-aware User Sequence Representation

We firstly utilize the transformer encoder to map high-dimensional, one-hot encoded item vectors into lower-dimensional, dense vectors using an embedding matrix $E \in \mathbb{R}^{|V| \times d}$ where d is the embedding size. A position matrix $P \in \mathbb{R}^{M \times d}$ is constructed for representing the position information of the sequences. Inspired by previous research[2], for an individual user $u_i \in U$, the behavior-aware embedding of its interacted item v_j is defined as:

$$h^0 = e_j + p_l + b_k, \tag{1}$$

where $e_j \in E$ represents the initialized embedding vectors of interacted item $v_j \in V$, $p_l \in P$ represents the learnable positional embedding of v_j and b_k represent the behavior type embedding. We denote the behavior-aware item embedding matrix of S_i as $H_i \in \mathbb{R}^{M \times d}$. And the representation of subsequence $S_{i,k}$ of S_i with behavior b_k is denoted as $H_{i,k}$ for simplicity.

4.3 Behavior-Aware Data Augmentation Based on Diffusion Models

Denoising diffusion probabilistic model (DDMP) is a type of generative model that typically involve two phases[14]: the forward process and the reverse process. For user u_i, we aim to get the augmented subsequence $S_{i,k}^{aug}$ from original subsequence $S_{i,k}$. Before the forward process begins, it is required to flatten $H_{i,k}$ in row-major order and convert the discrete input items into a continuous space. In the forward process, $H_{i,k}$ is iteratively added Gaussian noise in a Markov chain over T iterations. We denote $H_{i,k}^t$ as $H_{i,k}$ after t-th iteration. Assume that the original distribution is $H_{i,k}^0 \sim q(H_{i,k}^0)$, the forward process is formulated as follows:

$$q\left(H_{i,k}^t \mid H_{i,k}^{t-1}\right) = \mathcal{N}\left(H_{i,k}^t; \sqrt{1-\beta_t} h^{t-1}, \beta_t I\right), \tag{2}$$

where $\beta_t \in (0,1)$ represents the parameter that governs the scale of the Gaussian noise added at each timestep t. We adopt a linear noise schedule $\beta_t = \beta_0 + \frac{(\beta_T - \beta_0)t}{T}$, β_T and β_0 are the final and initial noise level, respectively.

The reverse process is designed to iteratively eliminate the added Gaussian noise of $H_{i,k}$. Inspired by [14], the simplified loss function of the model training is formulated as follows:

$$L_{simple} = \mathbb{E}_{t, H_{i,k}, \epsilon}\left[\left\|\epsilon_\theta\left(\sqrt{\bar{\alpha}_t} H_{i,k} + \sqrt{1-\bar{\alpha}_t}\epsilon, t\right) - \epsilon\right\|_2^2\right], \tag{3}$$

where $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$, ϵ is the random Gaussian Noise and ϵ_θ is for noise prediction.

To extract the weights of different behaviors, inspired by previous research[15] we apply the self-attentive mechanism formulated as:

$$w_k = \mathrm{Softmax}\left(\mathrm{MLP}\left(\mathrm{Tanh}\right)\left(H_i\right)\right), k = 1, ..., K. \tag{4}$$

We utilize a MLP to conduct the conditional denoising, which is formulated as:

$$\hat{H_{i,k}^0} = f_\theta\left(H_{i,k}^t, t, w_k^T H_{i,k}\right) = \mathrm{MLP}\left(H_{i,k}^t, t, w_k^T H_{i,k}\right). \tag{5}$$

Therefore, for the subsequence representation $H_{i,k}$, the loss of the reverse process of Equation (3) can be rewritten as follows:

$$\mathcal{L}_{u_{i,k}}(\theta) = \mathbb{E}_{t,H_{i,k}^0}\left[\left\|f_\theta\left(H_{i,k}^t, t, w_k^T H_{i,k}^0\right) - H_{i,k}^0\right\|_2^2\right], \tag{6}$$

where $H_{i,k}^0$ is the initial representation of subsequence $H_{i,k}$.

4.4 Behavior-Aware Contrastive Learning

Previous research shows that contrastive learning improves model robustness against noisy interactions in diffusion models[13][15]. In our framework, we apply contrastive learning to user subsequences u_i, minimizing differences between behavior representations from the same user while maximizing differences from other users' sequences.We detail our contrastive learning approach for $H_{i,k}$ below.

Intra-Behavior Self-Supervised Learning Sequence-level rule-based augmentation such as cropping, reordering, and masking is prevalent in sequential recommendation [11,16]. Inspired by the previous research [3,17], since the operation in the forward process is similar to noise injection or data insertion, we adopt a random mask[3,16] to create different views of the original subsequences. For $H_{i,k}$, we take a random mask of a proportion $\eta (0 < \eta < 1)$ for the items to create a masked view. If an item $h_{i,j}$ is selected, it will be substituted with the 'mask' token, otherwise $h_{i,j}^{mask} = h_{i,j}$.

Generally, views derived from the same sequences are regarded as positive pairs, whereas those from distinct sequences are treated as negative pairs. So the intra-behavior cross-entropy loss for $H_{i,k}$ is conducted as:

$$\mathcal{L}_{i,k}^{intra} == -\log \frac{\exp\left(\text{sim}\left(H_{i,k}^t, H_{i,k}^{mask}\right)\right)}{\sum_{u \in batch} \sum_{k',k''=1,k' \neq k''}^{K} \exp\left(\text{sim}\left(H_{i,k'}^t, H_{j,k''}^{mask}\right)\right)}, \tag{7}$$

where sim(.) is an inner product of vectors to measure the similarity of two augmented views.

Inter-Behavior Self-Supervised Learning For each user u_i, a positive pair for contrastive learning is constructed from augmented views of an auxiliary behavior and the target behavior, while negative pairs are formed from augmented views of two auxiliary behaviors from different users. The inter-behavior cross-entropy loss is defined as:

$$\mathcal{L}_{i,k}^{inter} = -\log \frac{\exp\left(\text{sim}\left(\tilde{H_{i,k}}, \tilde{H_{i,K}}\right)\right)}{\sum_{u \in batch, i \neq j} \sum_{k' \neq k''} \exp\left(\text{sim}\left(\tilde{H_{i,k'}}, \tilde{H_{j,k''}}\right)\right)}, \tag{8}$$

where $\tilde{H_{i,k}} = v(H_{i,k}), s.t.v \sim \mathcal{V}$. $\mathcal{V}$ is a set of pre-defined data transformation function encompassing normal rule-based augmentation.

Multi-Task Joint Optimization According to the Equation (6)(7)(8), for each $H_{i,k}$, the joint optimization in the t-th iteration is:

$$\mathcal{L} = \mathcal{L}_{u_{i,k}} + \alpha\mathcal{L}_{i,k}^{intra} + \beta\mathcal{L}_{i,k}^{iner}, \tag{9}$$

where α and β are the hyperparameters to control the weight of different contrastive learning. Using the trained model, continuous augmented embedding sequences are generated. For each embedding vector, we compute its cosine similarity with all item embeddings and select the item with the highest similarity, resulting in a discrete augmented sequence $S_{i,k}^{aug}$.

4.5 Behavioral Logical Data Check and Completion

In MBSR, the occurrence of behavior k often presupposes the occurrence of k'. Specifically, if user u_i interacts with item v_j in behavior k, then u_i must have interacted with item v_j in behavior k', but vice versa is not necessarily true. For example, in some e-commerce platforms, a user must click an item before purchasing it, but a click does not always lead to a purchase. To capture these relationships, we define behavioral logic chains to represent the dependencies between behaviors.

Definition 2. *Behavioral Logic Dependency. For behaviors b_k and $b_{k'}$ of user u_i, if u_i interacts with item v_j in behavior b_k, there must be interaction between u_i and v_j in behavior $b_{k'}$, then it is said that b_k is dependent on $b_{k'}$. The relation of dependency is denoted as $b_{k'} \succeq b_k$ or $b_k \preceq b_{k'}$.*

The relations of multi-behavioral dependencies like $b_k \preceq b_{k'} \preceq b_{k''}$... or $b_{k''} \succeq b_{k'} \succeq b_k$... constitute the behavioral logic chains. Obviously, if there is $b_{k'} \succeq b_k$ for user u_i, there must be $S_{i,k} \subseteq S_{i,k'}$. However, the data augmentation process may introduce inconsistencies that violate behavioral logic dependencies.For user u_i, if there is $(v_{i,j}, b_{i,j}) \in S_{i,k}^{aug}$ and $b_{k'} \succeq b_k$ but $(v_{i,j}, b_{i,j}) \notin S_{i,k'}^{aug}$, add $(v_{i,j}, b_{i,j})$ to $S_{i,k'}^{aug}$ to revise the augmented data to satisfy the behavioral logic dependencies. We search for all behavioral logic chains and check the augmented subsequences starting from the tail of the chains (behaviors on the leftmost side of $\preceq$ or the rightmost side of $\succeq$). If violations of behavioral logic dependencies are found, we perform behavioral data completion. This process repeats until all $(v_{i,j}, b_{i,j}) \in S_i^{aug}$ satisfy the behavioral logic dependencies. Finally, the augmented subsequences are combined according to the behavioral logic chains.

5 Experiments

5.1 Experiment Settings

Data Sets We conduct our experiments on two public multi-behavior datasets: Tmall[3] and JDdata[4].

[3] https://tianchi.aliyun.com/dataset/dataDetail?dataId=649
[4] https://jdata.jd.com/html/detail.html?id=8

Baselines of Data Augmentation We compare our method with the following item-level baseline methods in MBSR:(1) None. No augmentation, using the original datasets as is. (2) Randomly augmentation. Augment each sequence by randomly selecting items from the original sequence. (3) ASReP[7]. An item-level data augmentation framework. We use ASReP for all subsequences of users. (4)MBASR[3]. A data augmentation framework for MBSR.As described in the source paper [3], we use only click as the auxiliary behavior.

Backbone Recommendation Models We compare our method with the following 5 MBSR models: (1) RIB[20]. (2)BINN[19]. (3) BAR[18]. (4) MBHT[2]. (5) MBSTR[9].

Evaluation Metrics We adopt two common metrics, i.e., hit ratio (HR) and normalized discounted cumulative gain (NDCG). And we report the top-10 results for each test group.

5.2 Overall Performance

The results show that our DiffAMB framework consistently enhances performance across all baseline models and datasets, outperforming other data augmentation methods. Random augmentation generally underperforms, with only occasional minor improvements, and lacks consistency. ASReP's effectiveness is limited for most baselines due to suboptimal augmented data quality, and its application in multi-behavior scenarios worsens the negative impact of inappropriate augmentation, underscoring that not all augmentation methods are beneficial. While MBASR often improves recommendation performance, its reliance on only one auxiliary behavior limits its effectiveness compared to DiffAMB. Detailed performance statistics are provided in Table 1, with the highest scores in bold and the best baseline results underlined.

Table 1. Performance comparison of different methods on Tmall and JD datasets.

		RIB		BINN		BAR		MBHT		MBSTR	
DataSet	Augmentation	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
Tmall	None	0.0297	0.0169	0.0198	0.0102	0.0379	0.0252	0.0546	0.0325	0.0624	0.0371
	Random	0.0279	0.0158	0.0192	0.0098	0.0393	0.0249	0.0553	0.0328	0.0645	0.0374
	ASReP	0.0283	0.0142	0.0180	0.0088	0.0357	0.0346	0.0494	0.0293	0.0519	0.0348
	MBASR	0.0313	0.0172	0.0211	0.0105	0.0382	0.00253	0.0565	0.0339	0.0637	0.0376
	DiffAMB	**0.0329**	**0.0190**	**0.0213**	**0.0114**	**0.0398**	**0.0279**	**0.0601**	**0.0349**	**0.0690**	**0.0416**
JDdata	None	0.3272	0.1868	0.3255	0.1847	0.3338	0.1731	0.5226	0.2125	0.5433	0.2237
	Random	0.3334	0.1871	0.3272	0.1852	0.3307	0.1683	0.5242	0.2095	0.5548	0.2232
	ASReP	0.3057	0.1819	0.3209	0.1842	0.3113	0.1653	0.5248	0.1996	0.5190	0.2134
	MBASR	0.3286	0.1912	0.3269	0.1865	0.3531	0.1868	0.5537	0.2207	0.5605	0.2262
	DiffAMB	**0.3476**	**0.1944**	**0.3455**	**0.1946**	**0.3634**	**0.1875**	**0.5632**	**0.2219**	**0.5949**	**0.2373**

5.3 Ablation Study

We evaluate three variants of DiffAMB ("w/o SSL", "w/o SSL_{intra}" and "w/o SSL_{inter}") against the full framework on three MBSR backbone models: RIB, MBHT, and MBSTR. Removing all contrastive learning ("w/o SSL") results in excessive noise in subsequences due to the lack of multi-behavior constraints, severely degrading performance—even worse than no augmentation. Removing inter-contrastive learning ("w/o SSL_{inter}") also reduces performance due to noise and the cumulative effect of multi-behaviors. Removing intra-contrastive learning ("w/o SSL_{intra}") causes a slight performance drop compared to DiffAMB but remains comparable to or slightly better than no augmentation. Detailed results are shown in Table2.

Table 2. Ablation study on different contrastive learning.

		RIB		MBHT		MBSTR	
DataSet	Mehords	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
Tmall	None	0.0297	0.0169	0.0546	0.0325	0.0624	0.0371
	w/o SSL	0.0164	0.0097	0.0382	0.0216	0.0485	0.0252
	w/o SSL_{intra}	0.0305	0.0172	0.0562	0.0327	0.0647	0.0388
	w/o SSL_{inter}	0.0238	0.0146	0.0507	0.0306	0.0586	0.0366
	DiffAMB	**0.0329**	**0.0190**	**0.0601**	**0.0349**	**0.0690**	**0.0416**
JDdata	None	0.3272	0.1868	0.5226	0.2125	0.5433	0.2237
	w/o SSL	0.2315	0.1282	0.3238	0.1299	0.3897	0.1642
	w/o SSL_{intra}	0.3267	0.1856	0.5389	0.2188	0.5622	0.2306
	w/o SSL_{inter}	0.2672	0.1569	0.3982	0.1741	0.4774	0.1910
	DiffAMB	**0.3476**	**0.1944**	**0.5632**	**0.2219**	**0.5949**	**0.2373**

5.4 Analysis of Hyperparameter

We tune the coefficients α and β as defined in Equation(9) within the range [0.1, 0.2, 0.4, 0.6, 0.8, 1.0]. For α, it is observed that the optimal value of α is usually about 0.2-0.4. This indicates that intra-behavior contrastive learning enhance recommendation to some degree, but its effectiveness may diminish potentially due to the disruption of sequence relationships. For β, it is observed that HR@10 and NDCG@10 typically increase when β rises at about 0.8. This suggests that inter-behavior contrastive learning capture more robust representations of various behaviors. However, higher β values risk overfitting.

5.5 Analysis of Infrequent Interacting Users

To evaluate DiffAMB's impact on low-interaction users, we filter users from the Tmall dataset with target behavior sequences of length less than 20, categorizing them into three groups [3, 5], (5, 10], and(10, 20]. Experiments across backbone MBSR models show augmented sequences consistently outperform non-augmented ones. See Fig. 2 for details.

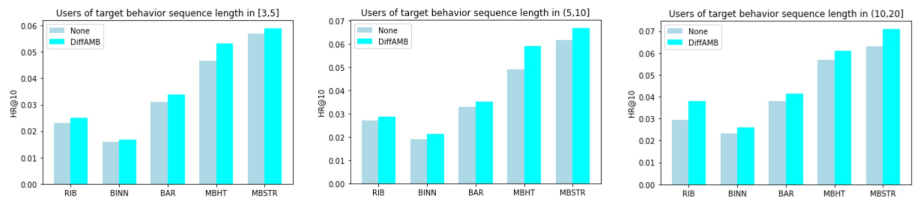

Fig. 2. Analysis of Infrequent Users with Interaction Sequence Lengths in [3, 20].

6 Conclusions

We propose DiffAMB, a diffusion-based framework for multi-behavior sequential recommendation, combining diffusion models and contrastive learning to generate high-quality, logically consistent augmented sequences. Experiments show DiffAMB addresses data imbalance and outperforms traditional methods, offering a novel solution for data sparsity and cold-start issues.

Acknowledgments. This work was supported by the National Key Research and Development Program of China under grant number 2023YFC3305704, the National Social Science Fund of China under grant number BCA240050 and the National Natural Science Foundation of China under grant number 62377015.

References

1. He Bowei, Xu He, Yingxue Zhang, Ruiming Tang, and Chen Ma: Dynamically expandable graph convolution for streaming recommendation. In: Proceedings of the ACM Web Conference 2023, pp. 1457-1467 (2023)
2. Yang Y., Huang C., Xia L., et al.: Multi-Behavior Hypergraph-Enhanced Transformer for Sequential Recommendation. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 2263-2274 (2022)
3. Zhou P, Huang Y, Xie Y, Gao J, Wand S, Kim J, Kim S: Is Contrastive Learning Necessary? A Study of Data Augmentation vs Contrastive Learning in Sequential Recommendation. In: Proceedings of the ACM Web Conference 2024(WWW 2024), pp. 3854-3863 (2024)
4. He Bowei, Xu He, Renrui Zhang, Yingxue Zhang, Ruiming Tang, and Chen Ma.: Dynamic Embedding Size Search with Minimum Regret for Streaming Recommender System. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, pp. 741-750 (2023)
5. He Bowei, and Chen Ma.: Interpretable Triplet Importance for Personalized Ranking. In Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, pp. 809-818 (2024)
6. Jing Xiao, Weike Pan, and Zhong Ming: A Generic Behavior-Aware Data Augmentation Framework for Sequential Recommendation. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1578 - 1588(2024)

7. Zhiwei Liu, Ziwei Fan, Yu Wang, and Philip S. Yu: Augmenting sequential recommendation with pseudo-prior items via reversely pre-training transformer. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1608–1612 (2021)
8. Qidong Liu, Fan Yan, Xiangyu Zhao, Zhaocheng Du, Huifeng Guo, Ruiming Tang, Feng Tian: Diffusion Augmentation for Sequential Recommendation. In: Proceedings of the 2023 ACM International Conference on Information and Knowledge Management, pp. 1576-1586 (2023)
9. Enming Yuan, Wei Guo, Zhicheng He, Huifeng Guo, Chengkai Liu, Ruiming Tang: Multi-Behavior Sequential Transformer Recommender. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1642-1652 (2022)
10. Juyong Jiang, Peiyan Zhang, Yingtao Luo, Chaozhuo Li, Jaeboum Kim, Kai Zhang, Senzhang Wang, Sunghun Kim: Improving Sequential Recommendations via Bidirectional Temporal Data Augmentation with Pre-training. arXiv preprint arXiv: 2112.06460 (2024)
11. Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Jiandong Zhang, Bolin Ding, Bin Cui: Contrastive Learning for Sequential Recommendation. In: IEEE 38th International Conference on Data Engineering, pp. 1259-1273 (2022)
12. Binquan Wu, Yu Cheng, Haitao Yuan, Qianli Ma: When Multi-Behavior Meets Multi-Interest: Multi-Behavior Sequential Recommendation with Multi-Interest Self-Supervised Learning. In: IEEE 40th International Conference on Data Engineering, pp. 845-858 (2024)
13. Cui, Ziqiang, Haolun Wu, Bowei He, Ji Cheng, and Chen Ma: Context Matters: Enhancing Sequential Recommendation with Context-aware Diffusion-based Contrastive Learning. In Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, pp. 404-414 (2024)
14. Jonathan Ho, Ajay Jain, and Pieter Abbeel: Denoising diffusion probabilistic models. In: Conference on Neural Information Processing Systems, pp. 6840–6851 (2020)
15. Wuchao Li, Rui Huang, Haijun Zhao, Chi Liu, Kai Zheng, Qi Liu, Na Mou, Guorui Zhou, Defu Lian, Yang Song, Wentian Bao, Enyun Yu, Wenwu Ou: DimeRec: A Unified Framework for Enhanced Sequential Recommendation via Generative Diffusion Models.arXiv preprint arXiv: 2408.12153 (2024)
16. Yizhou Dang, Enneng Yang, Yuting Liu, Guibing Guo, Linying Jiang, Jianzhe Zhao, Xingwei Wang: Data Augmentation for Sequential Recommendation: A Survey. arXiv preprint arXiv: abs/2409.13545 (2024)
17. Yongjun Chen, Zhiwei Liu, Jia Li, Julian McAuley, and Caiming Xiong: Intent Contrastive Learning for Sequential Recommendation. In: Proceedings of the ACM Web Conference 2022, pp. 2172-2182 (2022)
18. Mingkai He, Weike Pan, Zhong Ming: BAR: Behavior-aware recommendation for sequential heterogeneous one-class collaborative filtering. In: Information Sciences, 608, pp. 881-899 (2022)
19. Qiang Liu, Shu Wu, and Liang Wang: Multi-behavioral sequential prediction with recurrent log-bilinear model. In: IEEE Transactions on Knowledge and Data Engineering, **29**(6), 1254–1267 (2017)
20. Meizi Zhou, Zhuoye Ding, Jiliang Tang, and Dawei Yin: Micro behaviors: A new perspective in e-commerce recommender systems. In: Proceedings of the 11th ACM International Conference on Web Search and Data Mining, pp.727–735 (2018)