

Contents lists available at [ScienceDirect](https://www.sciencedirect.com)

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

A Preference-driven Conjugate Denoising Method for sequential recommendation with side information

Xiaofei Zhu ^a, Minqin Li ^a, Zhou Yang ^b,* ^a College of Computer Science and Engineering, Chongqing University of Technology, Chongqing, 400054, China^b College of Computer and Data Science, Fuzhou University, Fuzhou, 350108, China

ARTICLE INFO

Keywords:

Sequential recommendation

Side information

User representation learning

ABSTRACT

Sequential recommendation with side information aims to predict users' preferred items based on user behavior sequences. Previous methods utilize attention mechanisms to capture user preferences from behavior sequences but often neglect individual behavioral variations, which introduce varying levels of frequency and random noise, thus compromising preference identification and integration. To address this issue, we propose a Preference-driven Conjugate Denoising Method (PCDM) for sequential recommendation with side information. The method employs a conjugate denoising transformer, consisting of a Fourier denoising module for frequency noise elimination and a variational inference module for random noise reduction, followed by a conjugate transformer that learns the user preference representations. Subsequently, it utilizes a preference-driven denoised fusion module to integrate the learned representations, aligning them with true user preferences while minimizing mixed noise interference. Experiments on four datasets, including Amazon Beauty, Sports, Toys, and Yelp, report average gains of 8.39% in Recall@10, 9.16% in Recall@20, 6.14% in NDCG@10, and 6.94% in NDCG@20 compared to the latest models.

1. Introduction

As an important task for recommendation systems, sequential recommendation aims to predict users' preferred items based on their behavior sequences (Liu et al., 2024; Ying et al., 2018). It is widely applied in e-commerce platforms (Liu et al., 2022), social media (Gan, Wang, Yi, & Gu, 2024; Tang, Yu, Rokicki, Ewerth, & Dietze, 2021), and various other scenarios (Ren et al., 2023; Setty, Anand, Mishra, & Anand, 2017; Tavakolpoursaleh, Schaible, & Dietze, 2019).

Early methods identify user preferences by learning the clicked item sequence (Hidasi, 2015; Xu, He, & Li, 2018). Hidasi (2015) enhance session-based recommendations using Recurrent Neural Networks (RNN) by learning sequential dependencies in short-term user behaviors. Chang et al. (2021) propose a graph neural network-based framework for sequential recommendation, modeling user-item interactions as a dynamic graph to capture long-range dependencies and evolving user preferences. Zhou et al. (2023) introduce a method to refine attention distribution in Transformer models, improving sequential recommendation by mitigating overfitting and achieving more balanced user preference representations.

These methods rely on the clicked items in behavior sequences, yet neglecting the side information of items, such as brand or region, thus struggle to fully explore user preferences (Kang & McAuley, 2018; Qiu, Huang, Yin, & Wang, 2022; Sun et al., 2019; Zhou et al., 2023). To address this issue, recent methods adopt attention mechanisms to fuse item and side information to

* Corresponding author.

E-mail addresses: zxf@cqut.edu.cn (X. Zhu), lmq@stu.cqut.edu.cn (M. Li), 200310007@fzu.edu.cn (Z. Yang).<https://doi.org/10.1016/j.ipm.2025.104174>

Received 18 December 2024; Received in revised form 3 March 2025; Accepted 26 March 2025

Available online 17 April 2025

0306-4573/© 2025 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

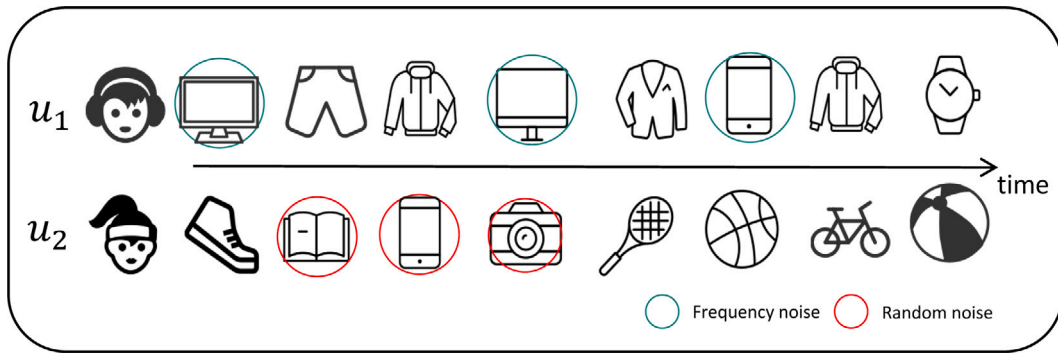


Fig. 1. Example illustration of frequency noise and random noise in user-item interaction histories. User 1, an electronics enthusiast, frequently browses electronics items but does not make purchases, indicating periodic behaviors that do not reflect true preferences (denoted by blue circles). These periodic behaviors introduce frequency noise, which can obscure the user's true interests. User 2, a sports enthusiast, occasionally clicks on unrelated items (e.g., camera) due to accidental clicks or exploratory browsing, introducing random noise (denoted by red circles). These random behaviors do not align with the user's true preferences and can mislead the recommendation model. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

enhance recommendation performance (Hao et al., 2023; Li et al., 2022; Lin et al., 2024; Liu et al., 2021; Xie, Zhou, & Kim, 2022; Yuan, Duan, Tong, Shi, & Zhang, 2021; Zhou et al., 2020). Liu et al. (2021) apply self-attention mechanisms to fuse important side information. Yuan et al. (2021) construct heterogeneous graphs to associate items with their side information, and then employ attention mechanisms to integrate information. Zhou et al. (2020) combine self-supervised mechanisms and mutual information maximization to gradually fuse item and side information. Li et al. (2022) employ a coarse-to-fine self-attention framework to explicitly learn hierarchical sequential information. Xie et al. (2022) use a fusion method with decoupled item and side information matrices to integrate information. Hao et al. (2023) combine self-attention and contrastive learning to fuse item and side information. Lin et al. (2024) design a multi-sequence ensemble attention layer to unify the fusion process for item and side information. These methods primarily rely on user behavior information, i.e., clicked items with side information, and have achieved satisfactory results.

Nevertheless, the aforementioned methods emphasize leveraging user behavior information but overlook the noise inherent in such information (Ma et al., 2020; Yang et al., 2024), which hampers the accurate identification and integration of genuine user preferences. Specifically, (i) **User behaviors are prone to introduce noise, especially the frequency and random noise (As shown in Fig. 1)** (Liang, Krishnan, Hoffman, & Jebara, 2018; Zhang, Yao, Sun and Tay, 2019), **hindering accurate identification of true preferences.** Users often exhibit recurring patterns (e.g., daily news browsing, weekly shopping), generating signals that mask their real preferences. Previous methods overlook this, making it difficult to discern true preferences from frequency noise. Moreover, not all behaviors genuinely reflect user preferences. Some interactions may be random or circumstantial (e.g., accidental clicks, exploratory browsing), introducing irrelevant noise. Previous methods ignore this, making it challenging to accurately identify preferences from preference-unrelated random noise. (ii) **The varying degree of frequency and random noise across users further impedes the aggregation of their true preferences.** Users exhibit diverse preference patterns — some have periodic purchasing habits, while others shop based on personal inclinations. This leads to the noise being mixed in user behaviors to varying degrees. Previous methods neglect the heterogeneous noise when integrating user preferences, leading to suboptimal preference representations.

To address this issue, we propose a Preference-driven Conjugate Denoising Method (PCDM) for sequential recommendation with side information. PCDM employs a conjugate denoising transformer to learn preference representations and adopts a preference-driven contrastive fusion module to merge information, thereby better predicting items. **For the conjugate denoising transformer,** we use Fourier denoising and variational inference denoising modules to reduce frequency and random noise respectively, and employ a conjugate transformer to jointly learn the denoised preference representations. **For the preference-driven contrastive fusion module,** we utilize contrastive learning to bring the denoised preference representations closer to the true user preferences in the semantic space, while also pulling the learned preference representations away from the mixed noise. Additionally, by employing multi-loss optimization — including item prediction loss, side information prediction loss, and contrastive learning loss — the model comprehensively learns user preference representations, thereby enhancing recommendation performance. Based on the fused representations, we then predict users' next item.

To validate PCDM's effectiveness, we conducted experiments on four datasets, i.e., Amazon Beauty, Sports, Toys and Yelp. Experimental results show that PCDM outperforms SOTA methods. Further analysis reveals that PCDM achieves more robust and stable performance due to its powerful denoising and fusion capabilities.

In summary, our contributions are as follows:

- (i) We observe that individual behavioral differences lead to varying levels of frequency and random noise masking true user preferences, which limits existing methods' prediction accuracy.

- (ii) We propose a conjugate denoising transformer to learn denoised preference representations by filtering both frequency and random noise. Additionally, we employ a preference-driven contrastive method to integrate user preferences while separating the mixed noise.
- (iii) Extensive experiments on public datasets show our method consistently outperforms SOTA baselines with robust performance.

2. Research objectives

To clearly articulate the research objectives of this paper, we delineate the limitations of existing research, the problems to be addressed, and the contributions of our proposed approach.

Limitations of Existing Research. Existing studies employ various attention mechanisms to identify and integrate user preferences within user behavior sequences. However, these methods overlook individual behavioral differences, which introduce varying levels of frequency and random noise, thus compromising preference identification and integration.

Problems to be Addressed. The problem we face is how to reduce the impact of frequency noise and random noise in user behavior and accurately identify users' true preferences.

Contributions of Our Research. Our main contributions are as follows: (1) We propose a conjugate denoising transformer to learn denoised preference representations by filtering both frequency and random noise. Additionally, we employ a preference-driven contrastive method to integrate user preferences while separating the mixed noise. (2) We propose a conjugate denoising transformer to learn denoised preference representations by filtering both frequency and random noise. Additionally, we employ a preference-driven contrastive method to integrate user preferences while separating the mixed noise. (3) Extensive experiments on public datasets show our method consistently outperforms SOTA baselines with robust performance.

3. Literature review

Existing methods on sequential recommendation task can be broadly categorized into two groups: basic sequential recommendation methods and sequential recommendation methods with side information.

Sequential Recommendation. Early sequential recommendation approaches focus on learning user behavior sequences. [Hidasi \(2015\)](#) employ RNN to capture sequential dependencies in user sessions for improved recommendations. [Kang and McAuley \(2018\)](#) introduce self-attention to better capture long-term interests. [Sun et al. \(2019\)](#) further enhance recommendation performance through bi-directional self-attention. [Qiu et al. \(2022\)](#) address the representation degeneration problem in sequential recommendation by introducing a contrastive learning framework that enhances embedding diversity and robustness, leading to improved user preference modeling. [Zhou, Yu, Zhao, and Wen \(2022\)](#) replace the multi-head self-attention mechanism in the Transformer-based sequential recommendation model with a learnable filtering layer, effectively filtering out the noise in the sequential data. [Zhou et al. \(2023\)](#) propose a method that optimizes attention distribution in Transformer models, enhancing sequential recommendation by preventing overfitting and ensuring more balanced representations of user preferences.

Sequential Recommendation with Side Information. To better capture user preferences, researchers explore methods to integrate side information. [Liu et al. \(2021\)](#) identify the information invasion problem in early fusion, and propose to incorporate side information. [Yuan et al. \(2021\)](#) construct heterogeneous graphs to fuse items and their side information. [Zhou et al. \(2020\)](#) design four auxiliary self-supervised tasks to extract and fuse important elements from side information. [Hao et al. \(2023\)](#) and [Zhang, Zhao et al. \(2019\)](#) employ separate attention layers to capture coarse-to-fine interests from item sequences and side information. [Li et al. \(2022\)](#) introduce a coarse-to-fine sparse modeling approach to progressively model items and their side information. [Xie et al. \(2022\)](#) propose to decouple attention scores of items and their side information to enhance information fusion. [Lin et al. \(2024\)](#) design a multi-sequence ensemble attention layer to adaptively leverage intra- and inter-sequence interactions.

Previous methods overlook the impact of frequency and random noise introduced by individual behavioral differences, which mask true user preferences. In contrast, our work explicitly addresses this issue through a conjugate denoising transformer and preference-driven contrastive fusion, enabling more accurate preference identification and integration while minimizing mixed noise interference.

4. Method

4.1. Overview

In this paper, we propose a Preference-driven Conjugate Denoising Method (PCDM) for sequential recommendation with side information. As shown in [Fig. 2](#), PCDM consists of four modules, including the Embedding Module (in [Section 4.3](#)), the Conjugate Denoising Transformer (in [Section 4.4](#)), the Preference-driven Contrastive Fusion Module (in [Section 4.5](#)), and the Prediction Module (in [Section 4.6](#)).

4.2. Task formulation

Sequential recommendation with side information can be defined as: Given the user's clicked item sequence $I_u = [I_1, I_2, \dots, I_n]$ and its side information $X_u = [X_1, X_2, \dots, X_n]$, the task aims to predict the next item I_{n+1} based on these two types of information. Here, I_i and X_i denote the ID and side information of the i th item respectively. The side information $X_i = [x_{i_1}, x_{i_2}, \dots, x_{i_o}]$ represents the relevant features of the i th item I_i , such as brand, category, etc.

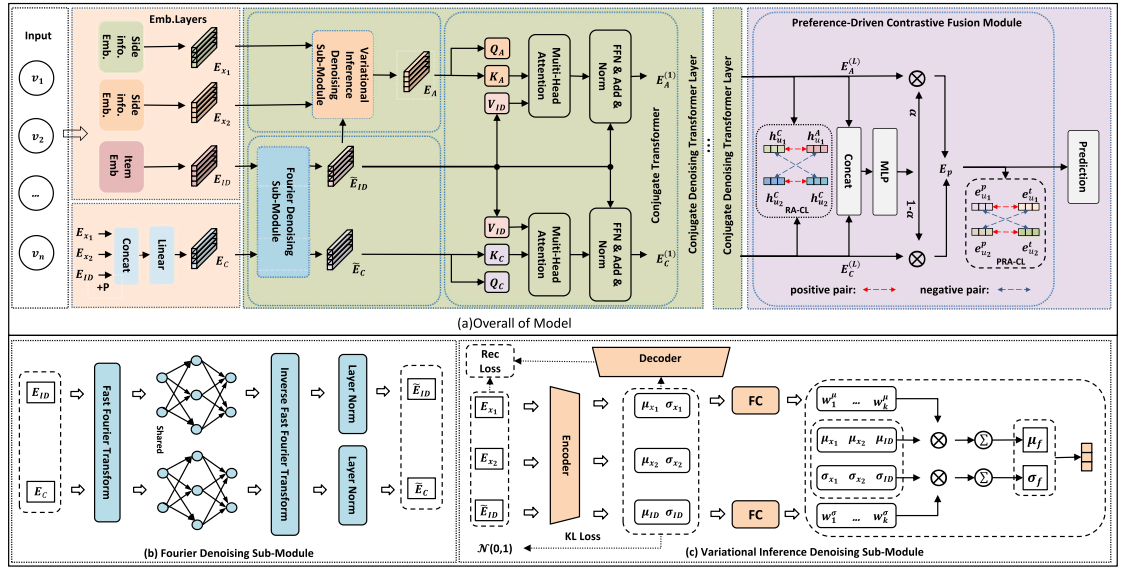


Fig. 2. Architecture of the Preference-Driven Conjugate Denoising Method (PCDM) for sequential recommendation with side information, which mainly consists of three key modules: (a) a Conjugate Denoising Transformer for frequency and random noise reduction and feature representation learning; (b) a Preference-driven Contrastive Fusion Module for feature representation integration and mixed noise reduction; (c) and a Prediction Module for the next item prediction.

4.3. Embedding module

To obtain representations of items and side information, we construct an embedding module. This module converts item ID sequences $I_u = [I_1, I_2, \dots, I_n]$ and side information $X_u = [X_1, X_2, \dots, X_n]$ into their corresponding embedding representations.

$$\begin{aligned} E_{ID} &= \Phi_{ID}([I_1, I_2, \dots, I_n]), \\ E_{X_i} &= \Phi_{X_i}([x_{i1}, x_{i2}, \dots, x_{io}]). \end{aligned} \quad (1)$$

where Φ_{ID} , Φ_{X_i} denote the conversion function. $i \in [1, m]$.

4.4. Conjugate denoising transformer

User behaviors often introduce substantial frequency and random noise into item and side information (Du et al., 2023; Zhou et al., 2022). Previous methods (Hao et al., 2023; Li et al., 2022; Lin et al., 2024; Liu et al., 2021; Xie et al., 2022; Yuan et al., 2021; Zhou et al., 2020) overlooked this noise when learning these representations, resulting in inaccurate information representations. To address this issue, we propose a conjugate denoising transformer. This module employs a Fourier denoising sub-module and a variational inference denoising sub-module to filter high-frequency and random noise from behavioral sequences, and uses a conjugate transformer to collaboratively learn information representations. Notably, our model contains L stacked conjugate denoising transformers. For ease of description, we will explain using the first layer as an example.

4.4.1. Fourier denoising sub-module

The application of the Fourier Transform (FFT) in denoising tasks leverages its advantages in frequency-domain analysis. According to sociological research (Du et al., 2023; Zhou et al., 2022), users often generate significant frequency noise through clicks and purchases. In our work, we apply 1D DFT to model the frequency characteristics of user behavior sequences rather than individual items. The transformation encodes periodic patterns, where low-frequency components correspond to long-term user preferences, while high-frequency components often represent noise (Shin, Choi, Wi, & Park, 2024). To avoid noise interference, we propose a Fourier denoising sub-module, inspired by the Fourier transform's advantage in reducing frequency noise. The proposed sub-module primarily consists of three steps: frequency domain transformation, frequency domain denoising, and frequency domain inverse transformation. By employing an FFT-based denoising module, we can better capture periodic patterns in user behavior sequences, thereby reducing the interference of frequency noise in user preference modeling.

Frequency Domain Transformation. Due to Fast Fourier Transform's (FFT) efficiency in frequency domain conversion (Han et al., 2024; Zhou et al., 2022), we use it to transform two feature representations, i.e., item ID and side information. For item ID, we directly input them into FFT for transformation. For side information, we concatenate it with ID representations before FFT input, since certain side information represents specific item attributes and exhibits strong dependencies with items. Formally, we have:

$$E_C = MLP(Concat(E_{X_1}, E_{X_2}, \dots, E_{X_m})) \quad (2)$$

$$H_{ID} = FFT(E_{ID}) \in \mathbb{C}^{n \times d}, H_C = FFT(E_C) \in \mathbb{C}^{n \times d} \quad (3)$$

where MLP is a linear layer, $Concat$ is a splice operation. $FFT(\cdot)$ denotes the one-dimensional FFT. H_{ID} and H_C represent the embedding of E_{ID} and E_C in the frequency domain, respectively, and they are a complex tensor. n and d represent the sequence length and the embedding size, respectively. \mathbb{C} denotes the complex space.

Frequency Domain Denoising. After the Fourier domain transform, we perform a filtering operation on each dimensional feature in the frequency domain by multiplying the spectrum with learnable weights.

$$\tilde{H}_{ID} = W \odot H_{ID}, \tilde{H}_C = W \odot H_C \quad (4)$$

where \odot is the element-level multiplication. $W \in \mathbb{C}^{n \times d}$ denotes the learnable weights.

Frequency Domain Inverse Transformation. We then convert the filtered frequency domain features back to the following time domain features:

$$F_{ID} \leftarrow FFT^{-1}(\tilde{H}_{ID}) \in \mathbb{R}^{n \times d} \quad (5)$$

$$F_C \leftarrow FFT^{-1}(\tilde{H}_C) \in \mathbb{R}^{n \times d} \quad (6)$$

where $FFT^{-1}(\cdot)$ denotes the inverse 1D FFT. F_{ID} and F_C represent the embedding of E_{ID} and E_C in the frequency domain after denoising, respectively.

Furthermore, we employ residual connections, layer normalization, and dropout techniques to stabilize model training.

$$\tilde{E}_{ID} = LayerNorm(E_{ID} + Dropout(F_{ID})) \quad (7)$$

$$\tilde{E}_C = LayerNorm(E_C + Dropout(F_C)) \quad (8)$$

where $LayerNorm(\cdot)$ is a layer normalization operation, and $Dropout(\cdot)$ is a random discard operation. \tilde{E}_{ID} and \tilde{E}_C are the outputs of the Fourier denoising sub-module.

4.4.2. Variational inference denoising sub-module

User behavior not only introduces frequency noise but also random noise (Han et al., 2024; Said, Jain, Narr, & Plumbaum, 2012). Regarding frequency noise, we have already explained how to handle it in the previous section. The remaining question is how to remove random noise. Regarding random noise, variational inference can reduce random noise by approximating the posterior distribution of clean data, thereby filtering out variations that deviate from this learned distribution (Chen & Guo, 2023; Im Im, Ahn, Memisevic, & Bengio, 2017). Its core idea is to learn an approximate distribution by maximizing the Evidence Lower Bound (ELBO), making it as close as possible to the true posterior distribution. In user behavior sequences, stochastic noise often manifests as incidental or exploratory actions unrelated to user preferences. By employing variational inference, we can construct latent space representations to capture stable user preference patterns, thereby filtering out random noise that deviates from these patterns.

Specifically, variational inference filters stochastic noise through the following steps: (1) Latent Space Representation Construction: A multilayer perceptron (MLP) is used to learn the mean and variance of item and side information, forming a normal distribution. (2) Representation Constraints: Reconstruction and stability constraints are imposed to ensure the accuracy and robustness of the latent space representations. (3) Representation Fusion: Different feature representations in the latent space are adaptively fused through weighted summation. The applicability of variational inference to user behavior sequences lies in its ability to capture stable preference patterns through latent space representations, thereby mitigating the interference of stochastic noise in preference modeling. By maximizing ELBO, variational inference ensures that the learned distribution better represents clean data, enhancing the model's denoising capability. Therefore, we design a variational inference denoising sub-module. This module includes three key steps: latent space representation construction, representation constraint, representation fusion.

Latent Space Representation Construction. To construct the latent space representations of items and side information, we use the Multilayer Perceptron (MLP) to learn the mean and variance of these features, thereby building normal distributions with corresponding parameters.

$$\mu_{ID} = MLP^\mu(\tilde{E}_{ID}), \sigma_{ID} = MLP^\sigma(\tilde{E}_{ID}); \quad (9)$$

$$\mu_{X_i} = MLP^\mu(E_{X_i}), \sigma_{X_i} = MLP^\sigma(E_{X_i}).$$

where $i \in [1, m]$, μ_{ID} and μ_{X_i} denote the mean value of the item and its associated side information, σ_{ID} and σ_{X_i} denote the variance of the item and its associated side information.

Based on the mean and variance, we apply the reparameterization trick to sample from the normal distributions, thus constructing the latent space representations of items and their side information.

$$\begin{aligned} Z_{ID} &= \mu_{ID} + \varepsilon \sigma_{ID}, \\ Z_{X_i} &= \mu_{X_i} + \varepsilon \sigma_{X_i}, \end{aligned} \quad (10)$$

where $i \in [1, m]$, ε is random noise sampled from a normal distribution $N(0, 1)$.

Representation Constraint. To ensure accuracy and stability of representations during training, we implement two constraints. The first is a reconstruction constraint, which reconstructs features from the latent space representation and requires the reconstructed representation to be as close as possible to the original. This constraint effectively prevents deviation of latent space representations during training.

$$\tilde{E}_{ID}^{rec} = MLP(Z_{ID}), E_{X_i}^{rec} = MLP(Z_{X_i}) \quad (11)$$

$$L_{rec} = \frac{1}{m+1} \left(\frac{\|\tilde{E}_{ID}^{rec} - \tilde{E}_{ID}\|_2^2}{d^*} + \sum_{i=1}^m \frac{\|E_{X_i}^{rec} - E_{X_i}\|_2^2}{d^*} \right) \quad (12)$$

where \tilde{E}_{ID}^{rec} is the reconstructed representation of Z_{ID} , $E_{X_i}^{rec}$ is the reconstructed representation of Z_{X_i} , $\|\cdot\|_2^2$ is the square of the L_2 paradigm, m denotes class m side information and d^* is the number of elements in the feature.

The second is a stability constraint. This constraint introduces Kullback–Leibler (KL) divergence to avoid excessive variance during training, thereby stabilizing the model training process.

$$L_{kl} = \frac{1}{m+1} (KL(N((\mu_{ID}, \sigma_{ID}) \| N(0, 1))) + \sum_{i=1}^m KL(N((\mu_{X_i}, \sigma_{X_i}) \| N(0, 1)))) \quad (13)$$

where $KL(\cdot)$ denotes the computation of KL scatter. $N(0, 1)$ is a normal distribution.

Representation Fusion. With the above approach, latent space representations of different features are obtained, i.e., item sequences (μ_{ID}, σ_{ID}) , side information $(\mu_{X_i}, \sigma_{X_i})$. To further integrate these information, we design learnable weights for each of them and adaptively fused them through weighted summation. This can be formulated as:

$$\mu = \sum w_a^\mu \mu_a, \sigma = \sum w_a^\sigma \sigma_a \quad (14)$$

$$E_A = \mu + \varepsilon \sigma^2, \varepsilon \sim N(0, 1) \quad (15)$$

where $a \in (ID, X_1, X_2, \dots, X_m)$, μ is the mean value of the item and its side information after mean fusion, σ is the variance of the item and its side information variance fused.

4.4.3. Conjugate transformer

To learn both frequency-denoised representation \tilde{E}_C and random-denoised representation E_A , we input both representations into a conjugate transformer $Con-Trans[Trans_C, Trans_A]$. In the conjugate transformer, we construct two sub-transformers with identical structures but different parameters: $Trans_C(Q_C, K_C, V_{ID})$, $Trans_A(Q_A, K_A, V_{ID})$. Each sub-transformer builds its own attention matrix with a unique perspective, using these matrices to aggregate the shared item representations. This multi-angle approach enables more comprehensive learning of user preferences.

Since both sub-transformers share the same structure, we will explain one of them for clarity. As with standard transformers, we first construct the attention matrix.

$$A_C = Attention_C(Q, K, V) = \text{soft max}\left(\frac{Q_C K_C^T}{\sqrt{d}}\right) V_{ID} \quad (16)$$

where Q_C and K_C are obtained by \tilde{E}_C through the linear layer. V_{ID} is obtained from \tilde{E}_{ID} by means of a linear layer. d is the hidden layer size.

We then build the feed-forward network and residual connections. Notably, our residual connections use the denoised item representations rather than the output from the previous transformer layer. This ensures the model maintains focus on the inherent item information.

$$L_C = LayerNorm(A_C + \tilde{E}_{ID}) \quad (17)$$

$$H_C = FFN_C(L_C) = ReLU(L_C W_1^C + b_1^C) W_2^C + b_2^C \quad (18)$$

where $LayerNorm(\cdot)$ is a layer normalization operation. $W_1^C, W_2^C, b_1^C, b_2^C$ are learnable parameters. $ReLU(\cdot)$ is an activation function. The H_C will be input to the next layer and become the E_C of the next layer.

4.5. Preference-driven contrastive fusion module

Users generate varying types of noise based on their behavioral patterns (He et al., 2023; Zhang, Chen, Zhao, Han, & Li, 2023). For example, some exhibit excessive high-frequency noise from periodic behaviors while others do not. Previous approaches (Hao et al., 2023; Xie et al., 2022) fused representations without considering individual user preferences, resulting in homogenized representations and compromised prediction accuracy. To overcome this limitation, we introduce a user preference-driven contrastive fusion approach. Our method leverages the ground-truth item (user's next click) as the true preference anchor and employs contrastive learning to align learned preferences with true preferences. This alignment ensures stronger semantic consistency between learned and true preferences, leading to more accurate preference representations.

Representation Alignment. Specifically, after inputting items and side information into an l -layer conjugate denoising transformer (in Section 4.4), we obtain frequency-denoised preference representation H_C and random-denoised preference representation H_A . To ensure consistency between these two representations, we design a contrastive learning loss. We treat representations of the same user (h_i^C, h_i^A) as a positive sample pair, and treat representations of different users (h_i^C, h_j^A) as negative sample pairs.

$$L_{RA-CL} = - \sum_{i=1}^B \log \frac{\exp(\text{sim}(h_i^C, h_i^A)/\tau)}{\sum \exp(\text{sim}(h_i^C, h_j^A)/\tau)} \quad (19)$$

where B denotes the batch size. $\text{sim}(\cdot)$ measures the similarity between two vectors. τ is a temperature hyperparameter.

Subsequently, we employ a gating mechanism to fuse the frequency-denoised representation H_C and random-denoised preference representation H_A to obtain the learned preference representation E_p .

$$\gamma = \phi(\text{Linear}(\text{Concat}(H_A, H_C))) \quad (20)$$

$$E_p = \gamma H_A + (1 - \gamma) H_C \quad (21)$$

where γ is a gate factor that varies with different users, which further extends the capability of representation learning. $\phi(\cdot)$ is the sigmoid activation function. Concat is the splicing operation. Linear is a linear layer.

To align the learned preference representation $e^p \in E_p$ with the true user preference representation e^t , we construct a contrastive loss. This loss guides the learned preference representation to be semantically closer to the true user preference while distancing it from mixed noise, thereby achieving better representation fusion

$$L_{PRA-CL} = - \sum_{i=1}^B \log \frac{\exp(\text{sim}(e_i^p, e_i^t)/\tau)}{\sum \exp(\text{sim}(e_i^p, e_j^t)/\tau)} \quad (22)$$

where e^t is the true user preference representation.

4.6. Prediction module

To better train the model, we employ multiple losses for optimization. Unlike previous single-loss approaches (Hao et al., 2023; Liu et al., 2021), this multi-loss method optimizes the model using both items and various side information, enabling more comprehensive and reliable model optimization.

$$\hat{y}_{X_i} = \phi(W_{X_i} E_p^T + b_{X_i}) \quad (23)$$

$$L_{SIL} = - \sum_{i=1}^m y_{X_i} \log(\hat{y}_{X_i}) + (1 - y_{X_i}) \log(1 - \hat{y}_{X_i}) \quad (24)$$

$$L_{ID} = - \sum_{i=1}^{|I|} y_i \log(\hat{y}_i) \quad (25)$$

where \hat{y}_{X_i} denotes the probability of attribute X_i , $\phi(\cdot)$ is the sigmoid activation function, W_{X_i} and b_{X_i} are the learnable parameters, and then we use the cross-entropy loss to compute L_{X_i} and L_{ID} . Overall, the total loss is defined as follows:

$$L_v = \lambda_1 L_{rec} + \lambda_2 L_{kl} \quad (26)$$

$$L_{cl} = \lambda_3 L_{RA-CL} + \lambda_4 L_{PRA-CL} \quad (27)$$

$$L_{total} = L_{ID} + L_{cl} + L_v + \lambda_5 L_{SIL} \quad (28)$$

where L_{ID} is the predicted loss of items, L_{SIL} is the predicted loss of side information, L_{cl} is contrasting learning loss, which contains L_{RA-CL} and L_{PRA-CL} . L_{rec} is the reconstruction loss, L_{kl} is the KL loss.

5. Experiments

5.1. Dataset

To valid the model performance, We perform experiments on four publicly available datasets: **Amazon Beauty**,¹ **Sports**,¹ **Toys**,¹ and **Yelp**.² The Amazon Beauty, Sports, and Toys datasets are derived from Amazon review data (McAuley, Targett, Shi, & Van Den Hengel, 2015). Consistent with the approach in prior work (Lin et al., 2024; Xie et al., 2022), we incorporate fine-grained product categories and position information as attributes for all three datasets. Yelp is a widely recognized dataset for business recommendation tasks. Consistent with the methodology in Lin et al. (2024) and Xie et al. (2022), we filter the dataset to include only transaction records dated after January 1, 2019. Additionally, business categories and geographical position information are incorporated as attributes in our experiments. Table 1 summarizes the statistics of these preprocessed datasets.

¹ <http://jmcauley.ucsd.edu/data/amazon/>.

² <https://www.yelp.com/dataset>.

Table 1

Statistics of the four real-world datasets (Beauty, Sports, Toys, and Yelp) after preprocessing. For each dataset, we include the number of users, items, and total actions (interactions), as well as the average number of actions per user and per item.

| Dataset | Beauty | Sport | Toys | Yelp |
|--------------|---------|---------|---------|---------|
| Users | 22,364 | 35,599 | 19,413 | 30,450 |
| Items | 12,102 | 18,358 | 11,925 | 20,039 |
| Actions | 198,502 | 296,337 | 167,597 | 317,182 |
| Avg.Act User | 8.9 | 8.3 | 8.6 | 10.4 |
| Avg.Act Item | 16.4 | 16.1 | 14.1 | 15.8 |

5.2. Baselines

In our experiments, we selected several representative sequential recommendation models from recent years as comparative baselines to demonstrate the effectiveness of PCDM. These include both sequential recommendation methods (SASRec, BERT4Rec, DuoRec, and AC-TSR) and sequential recommendation methods with side information (SASRecF, S3-Rec, ICAI, NOVA-SR, CAFE, DIF-SR, FDSA-CL, and MSSR). Among these, MSSR (Lin et al., 2024) is a recently published model.

To ensure a robust and fair evaluation, we selected three of the most competitive baseline models from recent advancements in sequential recommendation: NOVA-SR (Liu et al., 2021), DIF-SR (Xie et al., 2022), and MSSR (Lin et al., 2024). Experiments for these models were conducted under consistent settings, aligning datasets and key parameter configurations with those used in our proposed approach, enabling a direct and comparable analysis. For other baseline methods not included in our experiments, we relied on performance metrics reported in the respective literature. Additionally, since most mainstream baselines for this task do not use pre-trained models (Hao et al., 2023; Lin et al., 2024; Liu et al., 2021; Xie et al., 2022), we opted not to incorporate pre-trained models into our framework. Using pre-trained models can substantially boost performance, and including them would lead to an unfair comparison. Therefore, we chose to exclude pre-trained models to maintain fairness in our evaluations.

Sequential Recommendation Methods. SASRec (Kang & McAuley, 2018): Employs self-attention mechanism for sequential modeling. BERT4Rec (Sun et al., 2019): Utilizes bi-directional self-attention networks. DuoRec (Qiu et al., 2022): Addresses representation degradation through contrastive learning. AC-TSR (Zhou et al., 2023): Introduces attention calibration for Transformer-based sequential recommendation.

Sequential Recommendation Methods with Side Information. SASRecF: An extension of SASRec that incorporates item side information. S3-Rec (Zhou et al., 2020): Leverages contrastive learning with four pre-training tasks. FDSA-CL (Hao et al., 2023): Enhances FDSA (Zhang, Zhao et al., 2019) with contrastive learning at both item and side information levels. ICAI (Yuan et al., 2021): Integrates heterogeneous graph embeddings. NOVA-SR (Liu et al., 2021): Implements non-invasive attention mechanism. CAFE (Li et al., 2022): Focuses on next-item prediction. DIF-SR (Xie et al., 2022): Uses decoupled attention for fused representations. MSSR (Lin et al., 2024): Introduces multi-sequence integrated attention.

5.3. Implementation details

All experiments in this paper are conducted using Python 3.8 and the PyTorch 1.8.1 deep learning framework. The models are implemented within the RecBole framework (Zhao et al., 2021). We set the batch size and dropout rate to 512 and 0.5, respectively. The models are trained for 200 epochs using Adam optimizer. In addition, the learning rate on the four datasets is tested in $\{0.01, 0.001, 0.0001\}$, and the hidden layer size is set to 256. The weight $\lambda_1, \lambda_2, \lambda_3$ and λ_4 are selected from $\{0.1, 0.01, 0.001\}$, λ_5 is chosen from $\{5, 10\}$. Other hyperparameters follow the optimal settings from their original papers. All experiments are conducted on an NVIDIA GeForce RTX 3090 GPU.

5.4. Evaluation metrics

In our experiments, we use leave-one-out strategy for evaluation, following the prior works (Lin et al., 2024; Xie et al., 2022). For each user behavior sequence, we use the last two items for validation and testing respectively, while the other items are used for training. We evaluate the model performance using RECALL@K and NDCG@K metrics with overall ranking strategy.

5.5. Main results

Table 2 shows the overall results of baselines and our method across datasets. Among sequential recommendation methods, DuoRec (Qiu et al., 2022) outperforms other baselines on most metrics, mainly due to its effective information fusion through self-attention mechanism. For methods with side information, approaches using sophisticated fusion strategies (like attention) such as NOVA-SR (Liu et al., 2021), DIF-SR (Xie et al., 2022), and MSSR (Lin et al., 2024) outperform those using simple strategies (like feature addition) such as SASRecF and FDSA-CL (Hao et al., 2023). This demonstrates the necessity of well-designed fusion strategies.

Moreover, our method outperforms these baselines on most datasets. This is mainly because: (i) it removes two types of noise during feature learning, leading to better representations; (ii) it fuses features based on user preferences, resulting in more reasonable preference representations. However, our method underperforms on the Toys dataset. This is primarily due to the sparsity of the

Table 2

Performance comparison of different recommendation methods on four real-world datasets. For our proposed method (PCDM), the reported results are averaged over 10 independent runs to ensure robustness and statistical reliability. The baselines are divided into two categories: sequential recommendation (SR) methods and session-based intent-aware sequential recommendation (SISR) methods. Evaluation metrics include Recall@10,20 and NDCG@10,20. Bold and underlined numbers indicate the best and second-best performance.

| Dataset | Metric | SR baselines | | | | SISR baselines | | | | | | | | |
|---------|-----------|--------------|--------|--------|--------|----------------|---------|--------|--------|--------|---------|---------------|---------------|---------------|
| | | BERT4Rec | AC-TSR | SASRec | DuoRec | SASRecF | FDSA-CL | CAFE | S3Rec | ICAI | NOVA-SR | DIF-SR | MSSR | PCDM |
| Beauty | Recall@10 | 0.0529 | 0.0823 | 0.0828 | 0.0865 | 0.0719 | 0.0824 | 0.0840 | 0.0868 | 0.0879 | 0.0887 | 0.0891 | <u>0.0900</u> | 0.0939 |
| | Recall@20 | 0.0815 | 0.1227 | 0.1197 | 0.1251 | 0.1013 | 0.1115 | 0.1159 | 0.1236 | 0.1231 | 0.1237 | <u>0.1288</u> | 0.1282 | 0.1384 |
| | NDCG@10 | 0.0237 | 0.0373 | 0.0371 | 0.0441 | 0.0414 | 0.0424 | 0.0437 | 0.0439 | 0.0439 | 0.0439 | 0.0446 | <u>0.0448</u> | 0.0471 |
| | NDCG@20 | 0.0309 | 0.0474 | 0.0464 | 0.0539 | 0.0488 | 0.0497 | 0.0514 | 0.0531 | 0.0528 | 0.0527 | 0.0541 | <u>0.0544</u> | 0.0583 |
| Sport | Recall@10 | 0.0295 | 0.0548 | 0.0526 | 0.0483 | 0.0435 | 0.0447 | 0.0429 | 0.0517 | 0.0527 | 0.0534 | <u>0.0552</u> | 0.0547 | 0.0594 |
| | Recall@20 | 0.0465 | 0.0837 | 0.0773 | 0.0712 | 0.0640 | 0.0653 | 0.0611 | 0.0758 | 0.0762 | 0.0759 | 0.0809 | <u>0.0814</u> | 0.0871 |
| | NDCG@10 | 0.0130 | 0.0241 | 0.0233 | 0.0247 | 0.0235 | 0.0222 | 0.0254 | 0.0249 | 0.0243 | 0.0250 | 0.0257 | <u>0.0260</u> | 0.0282 |
| | NDCG@20 | 0.0173 | 0.0313 | 0.0295 | 0.0304 | 0.0286 | 0.0284 | 0.0299 | 0.0310 | 0.0302 | 0.0307 | 0.0321 | <u>0.0327</u> | 0.0351 |
| Toys | Recall@10 | 0.0533 | 0.0831 | 0.0831 | 0.0947 | 0.0733 | 0.0851 | 0.0809 | 0.0967 | 0.0972 | 0.0978 | 0.0994 | <u>0.1022</u> | 0.1029 |
| | Recall@20 | 0.0787 | 0.1208 | 0.1168 | 0.1297 | 0.1052 | 0.1169 | 0.1055 | 0.1349 | 0.1303 | 0.1322 | 0.1358 | <u>0.1402</u> | 0.1454 |
| | NDCG@10 | 0.0234 | 0.0375 | 0.0375 | 0.0487 | 0.0417 | 0.0417 | 0.0476 | 0.0475 | 0.0478 | 0.0480 | 0.0495 | 0.0508 | 0.0489 |
| | NDCG@20 | 0.0297 | 0.0470 | 0.0460 | 0.0575 | 0.0497 | 0.0507 | 0.0543 | 0.0571 | 0.0561 | 0.0567 | 0.0587 | <u>0.0604</u> | 0.0596 |
| Yelp | Recall@10 | 0.0524 | 0.0654 | 0.0650 | 0.0641 | 0.0413 | 0.0625 | 0.0633 | 0.0589 | 0.0663 | 0.0682 | 0.0690 | <u>0.0715</u> | 0.0858 |
| | Recall@20 | 0.0756 | 0.0939 | 0.0928 | 0.0951 | 0.0675 | 0.0921 | 0.0954 | 0.0902 | 0.0940 | 0.0991 | 0.1005 | <u>0.1032</u> | 0.1218 |
| | NDCG@10 | 0.0327 | 0.0401 | 0.0401 | 0.0378 | 0.0216 | 0.0377 | 0.0376 | 0.0338 | 0.0400 | 0.0413 | 0.0417 | <u>0.0428</u> | 0.0491 |
| | NDCG@20 | 0.0385 | 0.0473 | 0.0471 | 0.0450 | 0.0282 | 0.0451 | 0.0453 | 0.0416 | 0.0470 | 0.0491 | 0.0496 | <u>0.0507</u> | 0.0581 |

Table 3

Ablation study on different components of our model across four datasets using Recall@20 metric. W/O denotes removing the corresponding component: VIDSM (Variational Inference Denoising Sub-Module), FDSM (Fourier Denoising Sub-Module), CT (Conjugate Transformer), RA-CL (Contrastive Loss of Representation Align), PRA-CL (Contrastive Loss of Preference Representation Align), and SIL (Side Information Loss).

| Variants | Beauty | Sport | Toys | Yelp |
|------------|---------------|---------------|---------------|---------------|
| W/O VIDSM | 0.0982 | 0.0501 | 0.1417 | 0.1186 |
| W/O FDSM | 0.1244 | 0.0747 | 0.1364 | 0.0931 |
| W/O CT | 0.1354 | 0.0852 | 0.1433 | 0.1204 |
| W/O RA-CL | 0.1322 | 0.0807 | 0.1404 | 0.1154 |
| W/O PRA-CL | 0.1366 | 0.0796 | 0.1362 | 0.1201 |
| W/O SIL | 0.1353 | 0.0856 | 0.1437 | 0.1201 |
| PCDM | 0.1384 | 0.0871 | 0.1454 | 0.1218 |

dataset. For instance, the average number of item interactions in the Toys dataset is significantly lower than in other datasets, as shown in Table 1. The sparsity of the data makes it challenging for the model to extract robust preference patterns, thus causing it to misinterpret noise as user preferences and degrading its performance.

5.6. Ablation study

We conduct an ablation study to analyze the contribution of different modules in our model. Specifically, we introduce the following variants:

W/O VIDSM: The model removed the variational inference denoising sub-module (in Section 4.4.2).

W/O FDSM: The model removed the Fourier denoising sub-module (in Section 4.4.1).

W/O CT: The model without the conjugate transformer (in Section 4.4.3).

W/O RA-CL: The model without the contrastive loss of representation align (in Eq. (19)).

W/O PRA-CL: The model without the contrastive loss of preference representation align (in Eq. (22)).

W/O SIL: The model without the side information loss (in Eq. (25)).

As shown in Table 3, the variant models show performance degradation compared to the complete model, demonstrating that the design of each module is reasonable and effective. Specifically, after removing both denoising modules, the performance decreases substantially. This is mainly because without these modules, the model becomes more susceptible to both types of noise, making it difficult to accurately identify the true user preferences. Additionally, removing the conjugate transformer leads to a performance decline. This indicates that it is necessary to conduct collaborative learning of the two denoising representations. Furthermore, after removing the contrastive loss, the performance also declines to some extent. This suggests that preference-driven representation fusion is effective in promoting feature integration. Finally, removing the side information loss results in a slight decrease in metrics. This indicates that emphasizing side information can also facilitate the learning of user preferences to some degree.

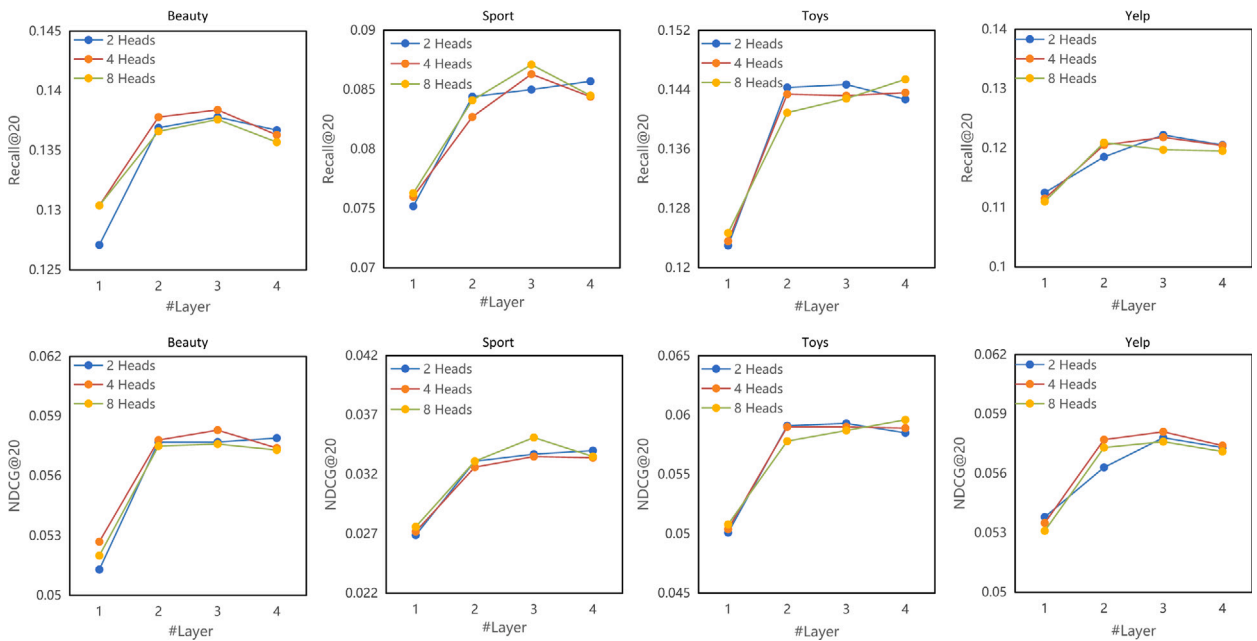


Fig. 3. These visualizations present the results of a parameter sensitivity experiment, showcasing the Recall@20 and NDCG@20 performance metrics for different neural network models across four datasets — Beauty, Sport, Toys, and Yelp. The graphs illustrate how the models' accuracy varies with changes in the number of layers (1, 2, 3, 4) and attention heads (2, 4, 8).

5.7. Impact of model complexity

We investigate the impact of model complexity by evaluating models with varying numbers of layers and attention heads. The experimental results are shown in Fig. 3. Specifically, we conduct experiments on four datasets (Beauty, Sports, Toys, and Yelp from left to right), using Recall@20 and NDCG@20 as evaluation metrics. In each subplot, the x-axis represents the number of layers (ranging from 1 to 4), while the y-axis show performance variations across different numbers of attention heads (2, 4, and 8). We observe that the model achieves strong performance even with relatively few layers and attention heads (e.g., 2 layers and 4 attention heads). This is primarily because the denoised model can more effectively learn true user preference representations, thus achieving better performance with fewer parameters.

5.8. Impact of fusion strategy

We compare our fusion module with two alternative fusion methods, e.g., Sum and Concat, to validate the effectiveness of our approach. As shown in Fig. 4, our preference-driven contrastive fusion module demonstrates consistently superior performance across multiple datasets. Notably, on the Beauty and Sport datasets, our method shows significant advantages over traditional Sum and Concat fusion strategies. These results strongly validate that the preference-driven contrastive fusion module can effectively enhance the model retrieval accuracy and ranking quality.

5.9. Effects of sequence length

We investigate the model performance under different sparsity conditions by varying the maximum sequence length (the sequence of user-clicked items). As shown in Tables 4 and 5, the performance exhibits subtle variations across different sequence lengths. The results indicate that the model performs best at the maximum sequence length. Despite longer sequences simultaneously introducing both informative signals and noise, our model maintains stable performance, indicating its ability to effectively filter noise while accurately identifying the true user preferences. Additionally, our model maintains stable performance even with shorter sequence lengths. This demonstrates the model's ability to effectively identify true user preferences under relatively sparse data conditions.

5.10. Impact of side information

To validate the impact of different side information, we conduct experiments on the Yelp² dataset using three types of side information (position, categories, and city) with NOVA, DIF-SR, and MSSR as baselines. As shown in Fig. 5, the incorporation of side information improves the performance of each model to varying degrees, with our method demonstrating the most

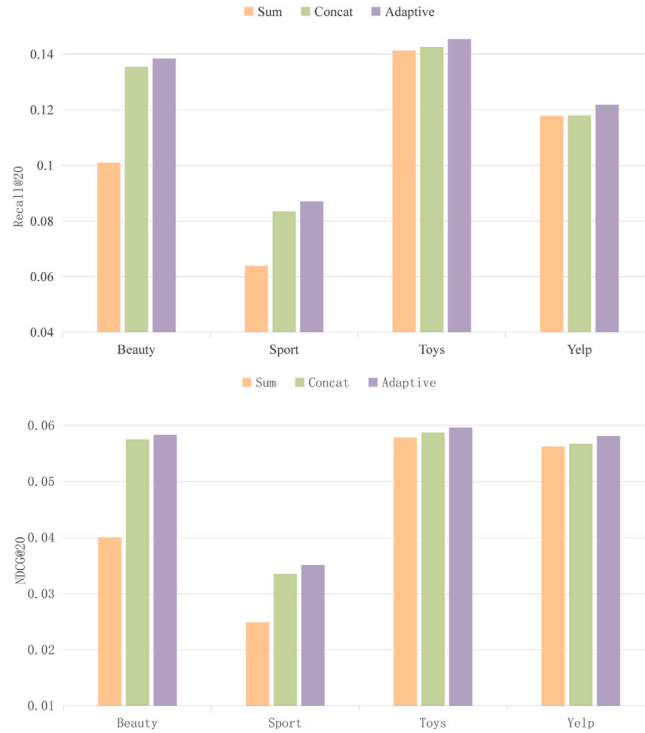


Fig. 4. Performance comparison of different feature fusion strategies (Sum, Concat, and Adaptive) across four datasets using Recall@20 (top) and NDCG@20 (bottom) metrics.

Table 4

Performance of the model for different sequence lengths (RECALL). The maximum length of the sequence takes the value 10, 20, 30, 40, 50.

| seq_len | Beauty | | Sport | | Toys | | Yelp | |
|---------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| | @10 | @20 | @10 | @20 | @10 | @20 | @10 | @20 |
| 10 | 0.0931 | 0.1369 | 0.0576 | 0.0860 | 0.1023 | 0.1427 | 0.0824 | 0.1170 |
| 20 | 0.0947 | 0.1368 | 0.0576 | 0.0858 | 0.1038 | 0.1449 | 0.0843 | 0.1204 |
| 30 | 0.0917 | 0.1350 | 0.0575 | 0.0851 | 0.1022 | 0.1432 | 0.0849 | 0.1206 |
| 40 | 0.0937 | 0.1363 | 0.0583 | 0.0860 | 0.0985 | 0.1389 | 0.0848 | 0.1227 |
| 50 | 0.0939 | 0.1384 | 0.0594 | 0.0871 | 0.1029 | 0.1454 | 0.0858 | 0.1218 |

Table 5

Performance of the model for different sequence lengths (NDCG). The maximum length of the sequence takes the value 10, 20, 30, 40, 50.

| seq_len | Beauty | | Sport | | Toys | | Yelp | |
|---------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| | @10 | @20 | @10 | @20 | @10 | @20 | @10 | @20 |
| 10 | 0.0473 | 0.0583 | 0.0271 | 0.0343 | 0.0484 | 0.0586 | 0.0479 | 0.0566 |
| 20 | 0.0470 | 0.0576 | 0.0272 | 0.0344 | 0.0492 | 0.0596 | 0.0481 | 0.0572 |
| 30 | 0.0464 | 0.0573 | 0.0271 | 0.0341 | 0.0492 | 0.0596 | 0.0490 | 0.0580 |
| 40 | 0.0467 | 0.0574 | 0.0273 | 0.0343 | 0.0468 | 0.0570 | 0.0489 | 0.0584 |
| 50 | 0.0471 | 0.0583 | 0.0282 | 0.0351 | 0.0489 | 0.0596 | 0.0491 | 0.0581 |

significant improvements. This indicates that additional side information facilitates more accurate identification of user preferences. Furthermore, it demonstrates the superior ability of our approach to utilize this information. The primary reason behind this effectiveness is the capability to effectively filter noise from side information.

5.11. Noise robustness of models

To evaluate noise robustness, we conduct comparative experiments with NOVA (Liu et al., 2021), DIF-SR (Xie et al., 2022), and MSSR (Lin et al., 2024) on the Beauty¹ and Yelp² datasets. We inject additive Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma^2)$ to the user representations, where σ controls the noise intensity. We test $\sigma \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$, corresponding to signal-to-noise ratios (SNR) from 20 dB

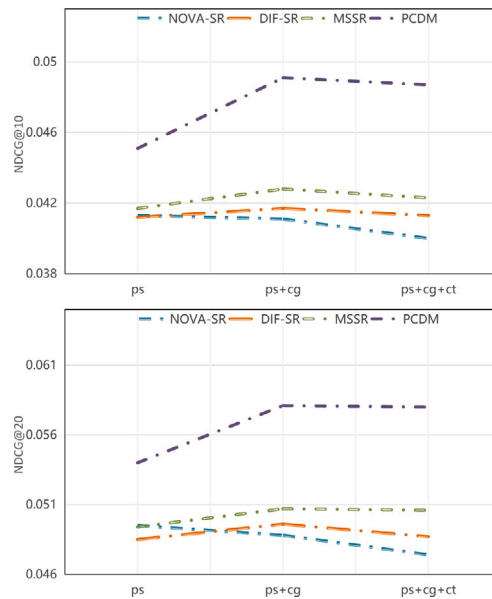


Fig. 5. Performance comparison of different models (NOVA-SR, DIF-SR, MSSR, and PCDM) when fusing different amounts of side information on the Yelp dataset, measured by NDCG@10 (top) and NDCG@20 (bottom). Three combinations of side information are evaluated: position information only (ps), position with category information (ps+cg), and position with category and city information (ps+cg+ct).

($\sigma = 0.1$) to 4 dB ($\sigma = 0.5$). Fig. 6 shows the results, with noise intensity on the x-axis and performance metrics on the y-axis. As noise intensity increases, all models show performance degradation, confirming the adverse effect of noise on recommendation quality. On the Beauty dataset, our model consistently achieves superior performance with a significant margin over the baseline models, particularly under high noise conditions. Similarly, for the Yelp dataset, our model exhibits robust performance with a more gradual degradation rate as noise intensifies, demonstrating enhanced noise resistance. These results demonstrate our model's superior robustness to noise.

Furthermore, we performed experiments to evaluate the performance of different recommendation models under the impact of two distinct types of noise: periodic and random. Periodic noise was introduced by inserting repeated items at regular intervals within the user behavior sequences, simulating cyclic or repetitive user behaviors. Specifically, for each sequence, items were inserted at fixed intervals (e.g., every 3rd, 4th, or 5th item). The experimental results are shown in Fig. 7. As the period length increases (i.e., the noise interval becomes larger), the performance degradation of the models becomes less pronounced. This indicates that high-frequency periodic noise (short periods) causes more interference to the models. Across different period lengths, PCDM consistently outperforms the baseline models, suggesting that the Fourier denoising module in PCDM effectively suppresses such noise. Random noise, on the other hand, was introduced by randomly replacing items within the sequences, simulating errors or missing data in real-world scenarios. The models were evaluated using Recall@20 and NDCG@20 metrics, with the horizontal axis representing different models (NOVA-SR, DIF-SR, MSSR, and PCDM) and the vertical axis representing different noise levels, expressed as percentages of the sequence affected by noise. The experimental results are shown in Fig. 8, where the performance of all models significantly deteriorates as the noise ratio increases, indicating that random noise has a substantial detrimental effect on the models. Under varying random noise ratios, PCDM consistently outperforms the baseline models, suggesting that the variational inference module of PCDM effectively alleviates this issue to some extent.

6. Theoretical and practical significance

We elucidate the significance of our work from both theoretical and practical perspectives.

Theoretical Significance. (1) This study introduces a Preference-Driven Conjugate Denoising Method (PCDM), utilizing a conjugate denoising transformer to address frequency and random noise in user behaviors, providing a novel perspective for noise handling and preference modeling in sequential recommendation tasks. (2) By incorporating Fourier denoising and variational inference modules, the conjugate transformer architecture achieves clearer preference representations in multi-noise environments, advancing theoretical understanding in sequential recommendation model research. (3) The proposed preference-driven contrastive fusion module aligns denoised preference representations with true user preferences while effectively filtering out mixed noise, creating new avenues for preference modeling in complex behavioral sequences.

Practical Significance. (1) Experimental results show that PCDM demonstrates robust and superior performance over current SOTA methods on multiple public datasets, including Amazon Beauty, Sports, Toys, and Yelp, underscoring its effectiveness in predicting user preferences in multi-noise contexts. (2) PCDM's denoising and fusion strategy enhances the robustness and accuracy

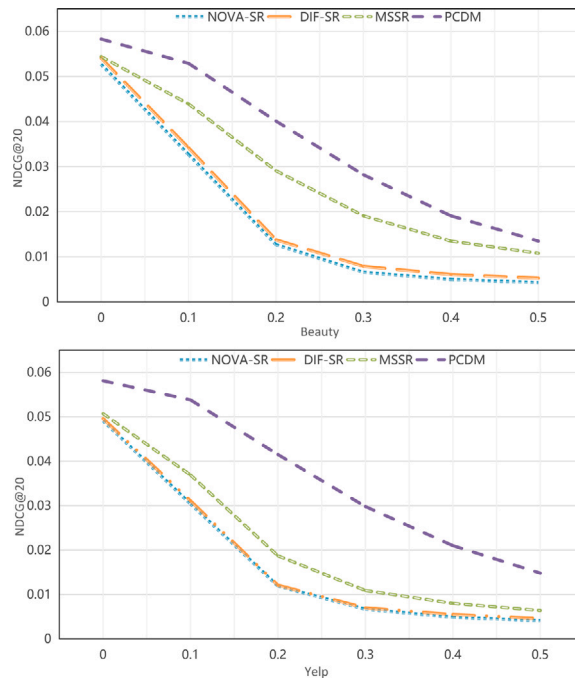


Fig. 6. Robustness analysis of different recommendation models (NOVA-SR, DIF-SR, MSSR, and PCDM) on Beauty and Yelp datasets under varying noise levels. The x-axis represents the noise ratio from 0 to 0.5, while the y-axis shows the NDCG@20 performance metric. PCDM demonstrates superior robustness by maintaining higher performance across increasing noise levels compared to baseline methods.

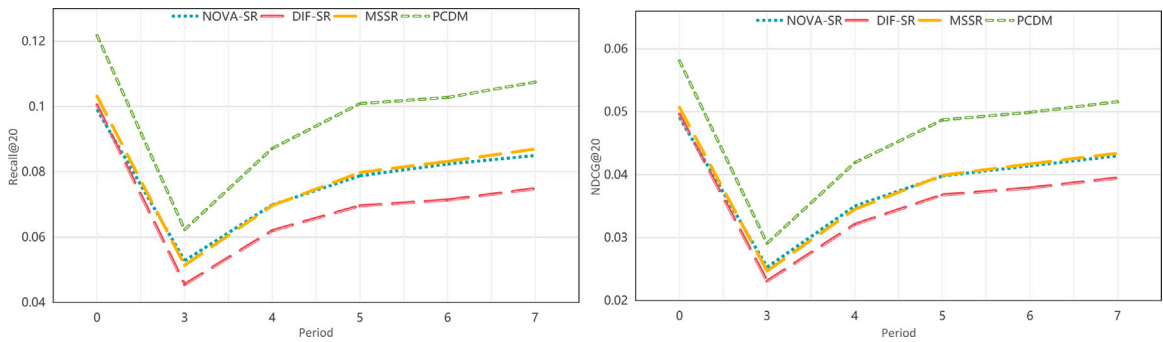


Fig. 7. Performance comparison of four models (NOVA-SR, DIF-SR, MSSR, and PCDM) under varying levels of periodic noise on the Yelp dataset. Noise is injected at different periodic intervals (period lengths 0, 3, 4, 5, 6, 7), with period 0 representing the baseline with no noise. The plots show the impact of periodic noise on Recall@20 (left) and NDCG@20 (right), highlighting the models' performance variations as the periodic noise increases.

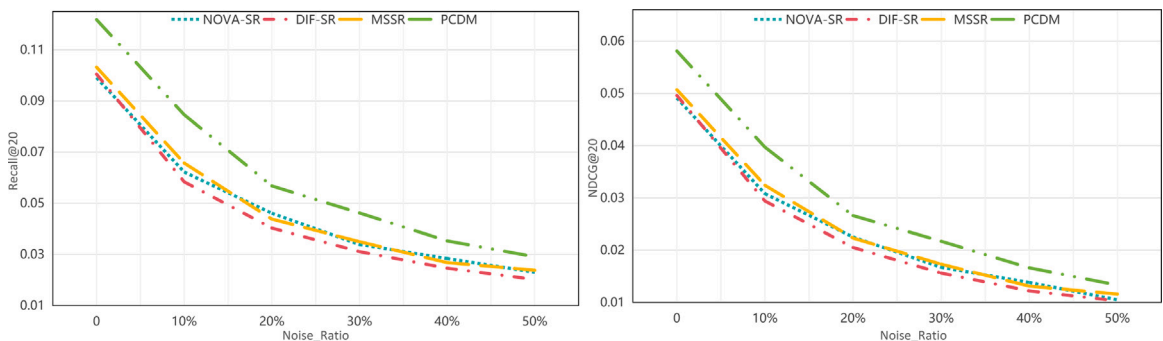


Fig. 8. Performance comparison of four models (NOVA-SR, DIF-SR, MSSR, and PCDM) under varying levels of random noise on the Yelp dataset. The noise ratio represents the percentage of items randomly replaced in user interaction sequences (0%, 10%, 20%, 30%, 40%, 50%). The plots show the impact of noise on Recall@20 (left) and NDCG@20 (right), demonstrating the models' robustness to increasing noise levels.

of recommendation systems, potentially improving user experience and extending the applicability of recommendation systems in complex environments. (3) This method provides a new technical approach for developing recommendation systems with noise-handling capabilities, promising to enhance recommendation diversity and personalization in fields such as e-commerce and social media.

7. Conclusion

In this paper, we have proposed a Preference-driven Conjugate Denoising Method (PCDM) for sequential recommendation with side information. We develop a conjugate denoising transformer to filter both frequency and random noise through Fourier denoising and variational inference denoising modules, and then propose a preference-driven contrastive fusion module to integrate user preferences while minimizing mixed noise interference. We conducted extensive experiments on four public datasets (Amazon Beauty, Sports, Toys, and Yelp), and the results demonstrate that the proposed method consistently outperforms state-of-the-art baselines with robust performance.

In the future, we will further explore the deeper impact of various types of noise on sequential recommendation with side information. Meanwhile, we will also investigate more effective fusion strategies.

CRedit authorship contribution statement

Xiaofei Zhu: Writing – review & editing, Supervision, Resources. **Minqin Li:** Writing – original draft, Methodology. **Zhou Yang:** Writing – review & editing, Supervision, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (62472059); the Natural Science Foundation of Chongqing, China (CSTB2022NSCQ-LZX0002, CSTB2022NSCQ-MSX1672); the Chongqing Talent Plan Project, China (CSTC2024YCJH-BGZX0022); the Major Project of Science and Technology Research Program of Chongqing Education Commission of China (KJZD-M202201102).

Data availability

Data will be made available on request.

References

- Chang, J., Gao, C., Zheng, Y., Hui, Y., Niu, Y., Song, Y., et al. (2021). Sequential recommendation with graph neural networks. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 378–387). <https://doi.org/10.1145/3404835.3462968>.
- Chen, S., & Guo, W. (2023). Auto-encoders in deep learning—a review with new perspectives. *Mathematics*, 11(8), 1777. <https://doi.org/10.3390/math11081777>.
- Du, X., Yuan, H., Zhao, P., Qu, J., Zhuang, F., Liu, G., et al. (2023). Frequency enhanced hybrid attention network for sequential recommendation. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval* (pp. 78–88). <http://dx.doi.org/10.1145/3539618.3591689>.
- Gan, M., Wang, C., Yi, L., & Gu, H. (2024). Exploiting dynamic social feedback for session-based recommendation. *Information Processing & Management*, 61(3), Article 103632. <https://doi.org/10.1016/j.ipm.2023.103632>.
- Han, Y., Wang, H., Wang, K., Wu, L., Li, Z., Guo, W., et al. (2024). Efficient noise-decoupling for multi-behavior sequential recommendation. In *Proceedings of the ACM on web conference 2024* (pp. 3297–3306). <http://dx.doi.org/10.1145/3589334.3645380>.
- Hao, Y., Zhang, T., Zhao, P., Liu, Y., Sheng, V. S., Xu, J., et al. (2023). Feature-level deeper self-attention network with contrastive learning for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(10), 10112–10124. <http://dx.doi.org/10.1109/TKDE.2023.3250463>.
- He, Z., Liu, W., Guo, W., Qin, J., Zhang, Y., Hu, Y., et al. (2023). A survey on user behavior modeling in recommender systems. arXiv preprint [arXiv:2302.11087](https://arxiv.org/abs/2302.11087). <https://doi.org/10.48550/arXiv.2302.11087>.
- Hidasi, B. (2015). Session-based recommendations with recurrent neural networks. arXiv preprint [arXiv:1511.06939](https://arxiv.org/abs/1511.06939). <https://doi.org/10.48550/arXiv.1511.06939>.
- Im Im, D., Ahn, S., Memisevic, R., & Bengio, Y. (2017). Denoising criterion for variational auto-encoding framework. vol. 31, In *Proceedings of the AAAI conference on artificial intelligence*. <http://dx.doi.org/10.1609/aaai.v31i1.10777>.
- Kang, W.-C., & McAuley, J. (2018). Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining* (pp. 197–206). IEEE, <http://dx.doi.org/10.1109/ICDM.2018.00035>.
- Li, J., Zhao, T., Li, J., Chan, J., Faloutsos, C., Karypis, G., et al. (2022). Coarse-to-fine sparse sequential recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval* (pp. 2082–2086). <http://dx.doi.org/10.1145/3477495.3531732>.
- Liang, D., Krishnan, R. G., Hoffman, M. D., & Jebara, T. (2018). Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference* (pp. 689–698). <http://dx.doi.org/10.1145/3178876.3186150>.
- Lin, X., Luo, J., Pan, J., Pan, W., Ming, Z., Liu, X., et al. (2024). Multi-sequence attentive user representation learning for side-information integrated sequential recommendation. In *Proceedings of the 17th ACM international conference on web search and data mining* (pp. 414–423). <http://dx.doi.org/10.1145/3616855.3635815>.

- Liu, C., Li, X., Cai, G., Dong, Z., Zhu, H., & Shang, L. (2021). Noninvasive self-attention for side information fusion in sequential recommendation. vol. 35, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 4249–4256). <http://dx.doi.org/10.1609/aaai.v35i5.16549>.
- Liu, B., Li, D., Wang, J., Wang, Z., Li, B., & Zeng, C. (2024). Integrating user short-term intentions and long-term preferences in heterogeneous hypergraph networks for sequential recommendation. *Information Processing & Management*, 61(3), Article 103680, <https://doi.org/10.1016/j.ipm.2024.103680>.
- Liu, D., Li, J., Wu, J., Du, B., Chang, J., & Li, X. (2022). Interest evolution-driven gated neighborhood aggregation representation for dynamic recommendation in e-commerce. *Information Processing & Management*, 59(4), Article 102982, <https://doi.org/10.1016/j.ipm.2022.102982>.
- Ma, J., Zhou, C., Yang, H., Cui, P., Wang, X., & Zhu, W. (2020). Disentangled self-supervision in sequential recommenders. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 483–491). <http://dx.doi.org/10.1145/3394486.3403091>.
- McAuley, J., Targett, C., Shi, Q., & 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* (pp. 43–52). <http://dx.doi.org/10.1145/2766462.2767755>.
- Qiu, R., Huang, Z., Yin, H., & 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* (pp. 813–823). <http://dx.doi.org/10.1145/3488560.3498433>.
- Ren, Z., Huang, N., Wang, Y., Ren, P., Ma, J., Lei, J., et al. (2023). Contrastive state augmentations for reinforcement learning-based recommender systems. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval* (pp. 922–931). <https://doi.org/10.1145/3539618.3591656>.
- Said, A., Jain, B. J., Narr, S., & Plumbaum, T. (2012). Users and noise: The magic barrier of recommender systems. In *Proceedings of the 32nd ACM conference on user modeling, adaptation and personalization* (pp. 237–248). https://doi.org/10.1007/978-3-642-31454-4_20.
- Setty, V., Anand, A., Mishra, A., & Anand, A. (2017). Modeling event importance for ranking daily news events. In *Proceedings of the tenth ACM international conference on web search and data mining* (pp. 231–240). <https://doi.org/10.1145/3018661.3018728>.
- Shin, Y., Choi, J., Wi, H., & Park, N. (2024). An attentive inductive bias for sequential recommendation beyond the self-attention. vol. 38, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 8984–8992). <http://dx.doi.org/10.1609/aaai.v38i8.28747>.
- Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., et al. (2019). BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management* (pp. 1441–1450). <http://dx.doi.org/10.1145/3357384.3357895>.
- Tang, R., Yu, R., Rokicki, M., Ewerth, R., & Dietze, S. (2021). Domain-specific modeling of user knowledge in informational search sessions. vol. 3052, In *CEUR workshop proceedings; 3052* (p. 8). Aachen, Germany: RWTH Aachen, <https://doi.org/10.15488/16879>.
- Tavakolpoursaleh, N., Schaible, J., & Dietze, S. (2019). Using word embeddings for recommending datasets based on scientific publications. In *LWDA* (pp. 365–370). http://ceur-ws.org/Vol-2454/paper_59.pdf.
- Xie, Y., Zhou, P., & Kim, S. (2022). Decoupled side information fusion for sequential recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval* (pp. 1611–1621). <http://dx.doi.org/10.1145/3477495.3531963>.
- Xu, J., He, X., & Li, H. (2018). Deep learning for matching in search and recommendation. In *Proceedings of the 41st international ACM SIGIR conference on research & development in information retrieval* (pp. 1365–1368). <https://doi.org/10.1145/3209978.3210181>.
- Yang, J., Ding, Y., Wang, Y., Ren, P., Chen, Z., Cai, F., et al. (2024). Debiasing sequential recommenders through distributionally robust optimization over system exposure. In *Proceedings of the 17th ACM international conference on web search and data mining* (pp. 882–890). <https://doi.org/10.1145/3616855.3635848>.
- Ying, H., Zhuang, F., Zhang, F., Liu, Y., Xu, G., Xie, X., et al. (2018). Sequential recommender system based on hierarchical attention network. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. <http://hdl.handle.net/10453/126040>.
- Yuan, X., Duan, D., Tong, L., Shi, L., & Zhang, C. (2021). Icai-sr: Item categorical attribute integrated sequential recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 1687–1691). <http://dx.doi.org/10.1145/3404835.3463060>.
- Zhang, C., Chen, R., Zhao, X., Han, Q., & Li, L. (2023). Denoising and prompt-tuning for multi-behavior recommendation. In *Proceedings of the ACM web conference 2023* (pp. 1355–1363). <http://dx.doi.org/10.1145/3543507.3583513>.
- Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, 52(1), 1–38, <https://doi.org/10.1145/3285029>.
- Zhang, T., Zhao, P., Liu, Y., Sheng, V. S., Xu, J., Wang, D., et al. (2019). Feature-level deeper self-attention network for sequential recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence* (pp. 4320–4326). <http://dx.doi.org/10.24963/ijcai.2019/600>.
- Zhao, W. X., Mu, S., Hou, Y., Lin, Z., Chen, Y., Pan, X., et al. (2021). Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms. In *Proceedings of the 30th acm international conference on information & knowledge management* (pp. 4653–4664). <http://dx.doi.org/10.1145/3459637.3482016>.
- Zhou, K., Wang, H., Zhao, W. X., Zhu, Y., Wang, S., Zhang, F., et al. (2020). S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM international conference on information & knowledge management* (pp. 1893–1902). <http://dx.doi.org/10.1145/3340531.3411954>.
- Zhou, P., Ye, Q., Xie, Y., Gao, J., Wang, S., Kim, J. B., et al. (2023). Attention calibration for transformer-based sequential recommendation. In *Proceedings of the 32nd ACM international conference on information and knowledge management* (pp. 3595–3605). <https://doi.org/10.1145/3583780.3614785>.
- Zhou, K., Yu, H., Zhao, W. X., & Wen, J.-R. (2022). Filter-enhanced MLP is all you need for sequential recommendation. In *Proceedings of the ACM web conference 2022* (pp. 2388–2399). <http://dx.doi.org/10.1145/3485447.3512111>.