Journal of Intelligent Information Systems (2024) 62:485–507
https://doi.org/10.1007/s10844-023-00825-w

RESEARCH

BMDF-SR: bidirectional multi-sequence decoupling fusion method for sequential recommendation

Aohua Gao[1,2] · Jiwei Qin[1,2] · Chao Ma[1,2] · Tao Wang[1,2]

Received: 14 July 2023 / Revised: 18 October 2023 / Accepted: 18 October 2023 /
Published online: 6 November 2023

Abstract

In the domain of sequence recommendation, contextual information has been shown to effectively improve the accuracy of predicting the user's next interaction. However, existing studies do not consider the dependencies between contextual information and item sequences, but the contextual information is directly fusing with the item sequences, which brings the problems described below: (1) Direct fusion fuses contextual information (e.g., time and categories) with item sequences which increases the dimensionality of the embedding matrix, thus increasing the complexity of the attention computation. (2) The attention computation of heterogeneous context information in the same embedding matrix makes it difficult for the recommendation model to distinguish this heterogeneous information. Therefore, we propose a bidirectional multi-sequence decoupling fusion method for sequence recommendation (BMDF-SR) to address the above issues. To establish the dependencies between temporal context sequences and item sequences, we first treat temporal contextual information as independent sequences and build bidirectional dependencies between contextual information sequences and item sequences via a three-layer seq2seq structure. Then, we perform attention computation independently for context sequences such as categories, and the complexity of attention computation can be effectively reduced by this decoupled attention computation. Moreover, since the attention computation is performed separately for each sequence, the interference between heterogeneous information during sequence fusion is reduced, allowing the model to effectively discriminate between different types of information. Extensive experiments on four real-world datasets show that the BMDF-SR method outperforms popular models.

✉ Jiwei Qin
jwqin@xju.edu.cn

Aohua Gao
gaoaohua@stu.xju.edu.cn

Chao Ma
chao@stu.xju.edu.cn

Tao Wang
xiaofan666@stu.xju.edu.cn

[1] School of Information Science and Engineering, Xinjiang University, Urumqi 830046, China

[2] Key Laboratory of Signal Detection and Processing, Xinjiang Uygur Autonomous Region, Xinjiang University, Urumqi 830046, China

Keywords Sequence recommendation · Bidirectional dependency · Multi-sequence model · Contextual information · Attention decoupling fusion

1 Introduction

Sequential recommendation(SR) has gained significant attention as a critical technique within recommendation systems(RS) (Lei et al., 2022; Xie et al., 2022). Among the traditional SR methods, Markov chain-based methods (e.g., FPMC Rendle et al., 2010) are widely used. Still, they infer user preferences from only one or a few recent user interactions (He et al., 2017), which makes it difficult for SR models to capture users' long-term preferences effectively. Recently, deep learning-based SR methods have received widespread attention, such as recurrent neural network (RNN)-based methods (e.g., GRU4Rec Hidasi et al., 2015 and HRNN Ling et al., 2018) and onvolutional neural network (CNN)-based methods (e.g., Caser Tang & Wang, 2018). These methods can capture a more comprehensive dependency relationship by exploiting users' long-term continuous interaction behavior (Zhang et al., 2023).

However, the above approaches do not distinguish the relevance of different user interactions to user preferences (Li et al., 2020). For example, users' purchasing and browsing behaviors in e-commerce platforms reflect different levels of user preferences for products. To distinguish the importance of different user interactions, researchers have incorporated attention mechanisms into SR models (e.g., SASRec Kang & McAuley, 2018, TiSASRec Li et al., 2020, BERT4Rec Sun et al., 2019). Recently, the research on improved SR methods based on attention mechanism mainly focuses on the introduction of additional supervision signals, which do not only emphasize the ID of the item, but enhance the performance of the recommendation model by accessing the context data (Zhong et al., 2023).

For instance, FDSA (Zhang et al., 2019) deploys two autonomous attention modules to merge item and contextual information, whereas S3-Rec (Zhou et al., 2019) capitalizes on contextual data within the pre-training phase. However, the above methods do not fully consider the dependency between the context information (such as time, category, etc.) and the item sequence, but choose to directly fusing with the item sequence (Xie et al., 2022), which brings the following problems: (1) To introduce more supervised signals, existing SR methods based on attention mechanisms introduce a large amount of auxiliary information, which leads to a continuously larger embedding matrix and thus increases the complexity of subsequent attention computations (Vaswani et al., 2017). The change in computational complexity is shown in Fig. 1. (2) Heterogeneous information from different sources (e.g., time, brand, category, etc.) is fused into the same embedding matrix for attention calculation, making it difficult for the model to distinguish heterogeneous information from different sources (Gong et al., 2023). Therefore, further improvement of the method is needed to better utilize the contextual information and address the above issues.

To solve the previously mentioned problems, we design a Bidirectional Multi-Sequence Decoupling Fusion method for sequence recommendation (BMDF-SR). We treat diverse contextual information (e.g., time, category, etc.) as independent sequences and use a seq2seq model to model the sequential dependencies within different sequences. Then, the bidirectional dependencies between temporal contextual information sequences and item sequences are modeled by alternatively overlaying the three-layer seq2seq model, which structure allows the model to handle noise better and missing data in the sequences (Sun et al., 2021). A critical component of BMDF-SR is the multi-sequence decoupling fusion (MDF) module, shown in

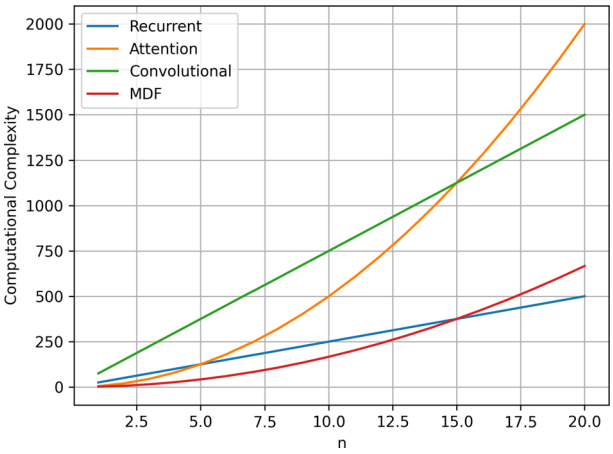

Fig. 1 Computational complexity variation graph. This figure represents the change in computational complexity of different deep learning methods at the same latitude with increasing sequence length

Fig. 2(b). Unlike traditional methods, MDF performs attention matrix computations before sequence fusion, and the attention matrix computation of each sequence is performed independently, which design significantly reduces the complexity of the attention computation (Vaswani et al., 2017), shown in Fig. 2(b). In addition, because each sequence's attention computation result is determined independently, interference between heterogeneous data during attention fusion is minimized, thereby enhancing the expressiveness of the attention matrix (Gong et al., 2023). As a result, our model can adeptly discriminate these heterogeneous information elements, resulting in improved RS performance.

It is worth mentioning that the MDF module is a versatile component that can replace the attention module in traditional SR methods based on attention mechanisms. The MDF module exhibits flexibility, allowing adaptive decoupling fusion of contextual information and item sequences based on different application scenarios. Such a design enables attention-

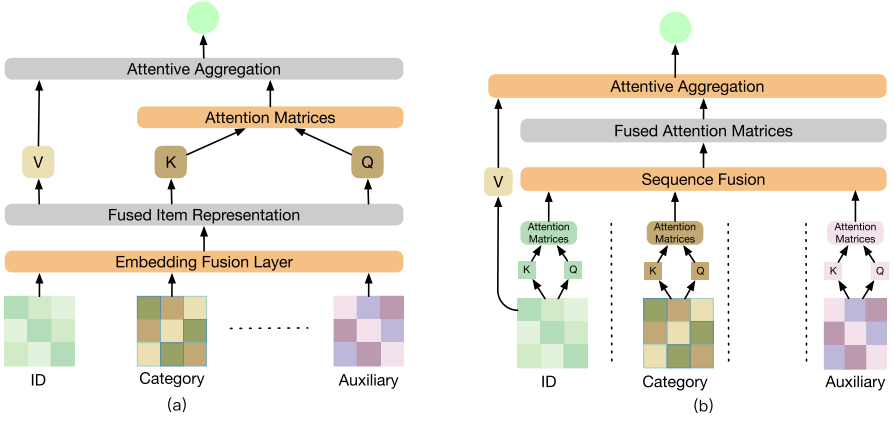

Fig. 2 Schematic diagram of the multi-sequence decoupling fusion module. In the figure, (a) indicates the traditional method of attentional fusion of contextual information, and (b) indicates the principle of the MADF module to fuse contextual information

based methods to better adapt to diverse application scenarios and provides a more flexible and scalable recommendation solution. The MDF opens new possibilities for improving personalized RS. In summary, the main contributions of our work can be summarized as follows:

- BMDF-SR treats contextual information (such as time, category, etc.) as separate sequences and employs a three-layer seq2seq structure to model the bidirectional dependencies between item and contextual information sequences. This design allows the model to better manage the presence of noisy and missing data within the sequences.
- The MDF module brings forward the attention calculation of contextual information sequences and item sequences, utilizing a decoupled approach for the attention matrix calculation. The MDF effectively reduces the complexity of the attention matrix computation and diminishes the interference between different heterogeneous information during the fusion of attention matrices. As a result, the model can effectively distinguish these heterogeneous information sources.
- BMDF-SR showcased outstanding performance on four real-world datasets. Furthermore, we carried out comprehensive ablation and parametric experiments to validate the efficacy and stability of the model.

2 Related work

In today's era of exponential information growth, RS have become essential in addressing information overload issues (Hao et al., 2023; Lei et al., 2022). Traditional RS are primarily based on content-based and collaborative filtering methods, which have proven effective (Tang et al., 2021). However, these traditional approaches mainly model user-item interactions in a static manner, focusing on capturing general user preferences (Duan et al., 2023; Lei et al., 2022). In practical applications, user preferences evolve with time (Ren & Gan, 2023), so modeling sequential dependencies within users' historical interactions is critical for accurate prediction, which is one of the main reasons SR has garnered significant attention in recent years (Zhou et al., 2023). By identifying sequential patterns in user behavior, SR methods can better comprehend the dynamic evolution of user interests and provide accurate recommendations corresponding to these patterns (Zhang et al., 2023). Consequently, SR has emerged as an essential and active area of research in RS.

Traditional SR methods often employ the Markov chain (MC) (Zhou et al., 2023; Garcin et al., 2013) concept to model user interaction sequences. For example, FPMC (Rendle et al., 2010) combines matrix factorization with MCs to model long-term and short-term preferences separately, while Fossil (Zhou et al., 2023) extends first-order MCs to higher orders by aggregating weighted and early interaction item representations. However, above traditional methods struggle to handle complex sequence patterns in today's scenarios (Garcin et al., 2013; Hidasi & Karatzoglou, 2018). To better learn user preferences, deep learning-based methods have been introduced into SR. For instance, Caser (Tang & Wang, 2018) embeds past interaction records into a vector space and extracts user behavior features for SR using horizontal and vertical convolutions. Although Caser (Tang & Wang, 2018) is a special case of MCs-based methods, like MCs-based methods, it fails to capture long-term dependencies of users, as MCs assume that the user's current state is only related to the most recent one or few states (Zhang et al., 2023). As a result, researchers began to use Recurrent Neural Networks (RNNs) to model sequential patterns in user interaction records. For example, GRU4Rec (Hidasi et al., 2015) utilizes Gated Recurrent Units (GRUs) to model

interaction sequences, while HRNN (Ling et al., 2018) designs a hierarchical RNN model to capture the evolution of user interests within a session. Although RNN-based SR methods can capture long-term dependencies by considering the complete user-item interaction sequence, they still suffer from the issue of neglecting the varying impact of different user behaviors on user preferences (Ye & Liu, 2022; Guo et al., 2023).

The attention mechanism, an essential method in deep learning, has been widely applied in the field of SR, which can assign different weights to each interaction in the input sequence to reflect their importance (Dang et al., 2023). For example, ATEM (Wang et al., 2018) selects relevant items in transaction-based RS by assigning weights to each observable item in a transaction. Recently, SASRec (Kang & McAuley, 2018) introduced the self-attention mechanism into the user-item interaction model, adopting a structure similar to the encoder part of the Transformer model. Building on SASRec (Kang & McAuley, 2018), FDSA (Zhang et al., 2019) separately models item sequences and auxiliary feature sequences using independent self-attention modules. BERT4Rec (Sun et al., 2019) employs a bidirectional encoder structure to learn latent representations in sessions and model the bidirectional dependencies within each session.

However, modeling only the order of item sequences has become inadequate for increasingly complex application scenarios (Xie et al., 2022). Consequently, researchers have continuously integrated auxiliary information into their models. For example, ATRank (Zhou et al., 2018) incorporates timestamps and item sequences, while TiSASRec (Li et al., 2020) converts time intervals into relative positions and combines them with item sequences (Liu et al., 2016; Yuan et al., 2020). Although these methods have achieved some success, they do not fully consider the dependencies between rich context information and item sequences, treating context information (such as time, category, brand, etc.) as auxiliary information (Wang et al., 2023). By fusing context information with item sequences during the embedding stage and representing them using attention calculations, it becomes challenging for models to distinguish these heterogeneous information sources (Xie et al., 2022). Additionally, the computational complexity of the attention mechanism increases with the rank of the embedding matrix (Vaswani et al., 2017). Therefore, effectively integrating various context information has become a critical issue in the current field of SR.

When improving attention-based methods, researchers have started to emphasize how to more effectively integrate auxiliary information (e.g., categories, timestamps,) with item sequences (Zhou et al., 2019; Liu et al., 2021; Yuan et al., 2021), rather than relying solely on item IDs as the model's sole input. Specifically, early methods like FDSA (Zhang et al., 2019) treated items and auxiliary information as two independent self-attention branches and then merged them in later stages. S3-Rec (Zhou et al., 2019) used self-supervised attribute prediction tasks during the pre-training phase to better handle auxiliary information. However, these methods often didn't fully consider the correlations between auxiliary information and items. Recently, some research has proposed a solution that involves directly embedding auxiliary information into item representations to achieve more precise attention mechanisms. For example, ICAI-SR (Yuan et al., 2021) used heterogeneous graph computation to embed items and categories before the attention layer, training item representations with incorporated category information in models with independent attribute sequences. Additionally, NOVA (Liu et al., 2021) introduced an approach that simultaneously provides pure item ID representations and integrated representations of auxiliary information to the attention layer. However, these methods didn't fully leverage auxiliary information to enhance item representation and prediction.

In recent years, several researchers have proposed various solutions in the field. For instance, Le et al. (2018) developed the Concurrent Basket Sequence (CBS) framework,

featuring a dual-network structure that enhances model performance by capturing the collaborative dynamics between support sequences and target sequences. Rakkappan and Rajan (2019) introduced the Stacked Temporal Context-Aware RNN method, which models item and context sequences separately through stacked RNNs. Zhang et al. (2023) proposed the TAT4Rec approach, utilizing an encoder-decoder structure to model time context sequences and item sequences individually while employing a window function module to preserve the continuous dependency relationships within the sequences.

3 Problem formulation

We define the set of all users $U = \{u_1, u_2, u_3, \ldots, u_n\}$ and the set of all items $I = \{i_1, i_2, i_3, \ldots, i_l\}$, where n represents the total number of users and l represents the total number of items. In the BMDF-SR problem, we define two sets: C represents the category context set and T represents the time context set. Specifically, within the item sequence, each item i has its own category context c_i. When a user interacts with item i at a specific time θ_i, we map θ_i to a 24-hour categorical feature space to represent the time context of item i. Therefore, for each user $u \in U$, there is a items sequence $S_u = \{i_1, i_2, i_3, \ldots, i_m\}$ arranged in chronological order, which corresponds to the category context sequence $S_c = \{c_1, c_2, c_3, \ldots, c_m\}$ and the time context sequence $S_t = \{t_1, t_2, t_3, \ldots, t_m\}$, where m represents the length of the sequence. Given the user's historical item sequence S_u, category context sequence S_c, and time context sequence S_t, our goal is to predict the next item that user u is most likely to interact with at time step $t_(m+1)$. In the embedding module of the model, the input sequences S_u, S_c, and S_t are passed through an embedding layer to obtain the item embedding sequence $S_u^e = \{i_1^e, i_2^e, i_3^e, \ldots, i_m^e\}$, category context embedding sequence $S_c^e = \{c_1^e, c_2^e, c_3^e, \ldots, c_m^e\}$, and time context embedding sequence $S_t^e = \{t_1^e, t_2^e, t_3^e, \ldots, t_m^e\}$.

We employ the principles of Neural Machine Translation (NMT) to establish a bidirectional dependency between the item embedding sequence S_u^e and the time context embedding sequence S_t^e Due to the phenomenon of semantic asymmetry between context and items, we introduce a Semantic Balance Module (SBM) to mitigate the issue of semantic asymmetry between context and items. The SBM can learn a compressed representation z_k of the input sequence, and after passing through the SBM module, a forward transformed sequence $Z = \{z_1, z_2, z_3, \ldots, z_m\}$ with semantic relative balance is generated. In the SBM module, we utilize a conventional loss function, formulated as follows:

$$\mathcal{L}_{SBM} = \mathbb{D}(q(z|x) \| p(z)) - \mathbb{E}_{q(z|x)}[log(p(x|z))]. \tag{1}$$

Where x represents the observable input data, z denotes the latent factor, $q(z|x)$ and $p(x|z)$ respectively represent the distribution of the inference model and the generative model. $\mathbb{D}(q \| p)$ represents the divergence between the two distributions, which employs relative entropy to measure the similarity between them. $\mathbb{E}$ can be regarded as the mathematical expectation of the reconstruction loss. In the MDF module, for each input sequence, let W_h denote the weight matrix of the h attention head, a_i^h represent the attention weight of the h head, and e_i denote the embedding vector of the i element of the input sequence. The output of the multi-head attention mechanism can be expressed as (2).

$$\mathcal{L}_i = \sum_{h=1}^{H} a_i^h W_h e_i \tag{2}$$

Where H represents the number of attention heads. To compute the loss function, we need to perform dot product operation between the output and the embedding vector of the target item, followed by normalization using the softmax function. Ren and Gan (2023) Let m_i denote the embedding vector of the target item. The loss function of the multi-head attention mechanism can be expressed as (3).

$$\mathcal{L}_{MDF} = -\frac{1}{N}\sum_{i=1}^{N}\frac{exp(\mathcal{L}_i \bullet m_i)}{\sum_{j=1}^{N} exp(\mathcal{L}_i \bullet m_j)} \tag{3}$$

Where N represents the number of elements in the sequence, and "$\bullet$" denotes the dot product operation. The objective is to minimize the loss function, allowing the model to effectively integrate the item sequence with other auxiliary context sequences.

4 Method

4.1 Overall architecture

Our model employs a three-layer seq2seq architecture, where each layer forms the sequential transformation relationship within its respective sequences. By stacking three layers of seq2seq modules, we effectively capture the bidirectional dependencies between the item and time context sequences (Sun et al., 2021). To further enhance the model's predictive capacity, we introduce a versatile MDF module after the first seq2seq layer to decouple and fusion various context sequences containing auxiliary information (e.g., time and brand). The MDF module allows for the decoupling and fusion of disparate context sequences and item sequences. Instead of merely concatenating these sequences for attention computation, the MDF module performs attention calculation in advance through a decoupling calculation method, obtaining attention matrices for each sequence before fusing them. This approach reduces computational complexity and increases the expressive power of the attention matrices, mitigating the impact of heterogeneous information on model performance (Xie et al., 2022; Vaswani et al., 2017). Moreover, it accounts for both the sequential transformation relationship in the item sequence and considering the sequential transformation relationship existing in the context sequence.

In our follow-up research, we identified semantic asymmetry between the time context and item sequences (Sun et al., 2021). Directly linking these sequences could lead to information loss and negatively affect model performance. We introduced an SBM before inputting the item sequence into the second seq2seq layer to address this issue. This module helps balance the semantic relationship between the time context and item sequences. Additionally, to tackle the influence of cold start and static preferences on the model's performance, we developed a PF (Preference Fusion) module that combines the user's static and dynamic preferences (Li et al., 2023). This integration enhances the prediction capability of a user's subsequent interaction behavior, ultimately improving the model's overall performance.

4.2 Multi-sequence decoupling fusion module

The MDF module, as depicted in Fig. 3, is comprised of multi-sequence decoupling attention calculation, attention fusion, and stacked feed-forward networks, which module receives two inputs: the hidden vector sequence $H^{(1)}$, obtained after the first layer of seq2seq, and the cat-

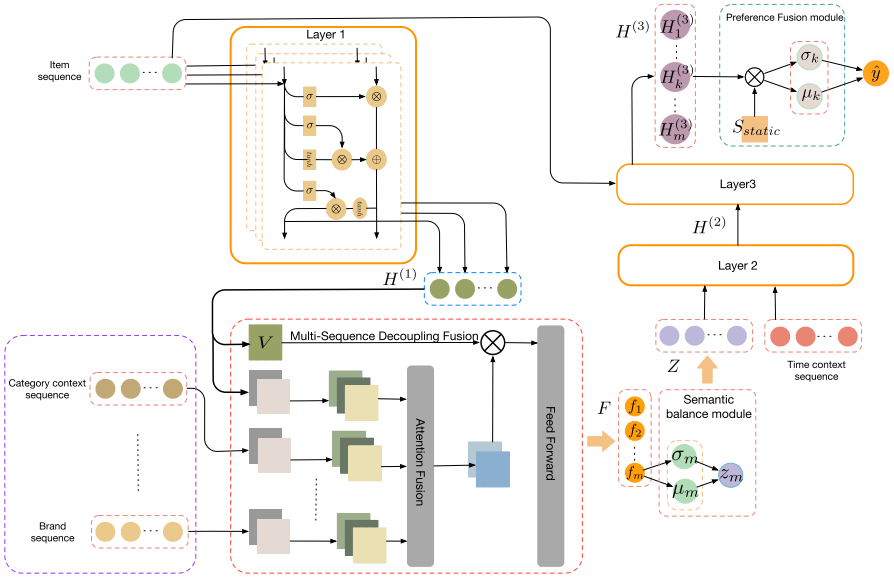

Fig. 3 The overall structure of BMDF-SR model. It mainly consists of a three-layer seq2seq module, an MDF module, a semantic balancing module and a fusion recommendation module

egory context embedding sequence S_c^e. Designed as a general module, it can merge multiple sequence representations. Specifically, we perform attention calculations on the sequences $H^{(1)}$, obtained through the forward transformation of the first layer of seq2seq, and the category context sequence S_e^c, resulting in their respective attention matrices $A_{multihead}^{S_u}$ and $A_{multihead}^{S_c}$. Subsequently, we employ an approach akin to the multi-head attention mechanism to fuse these two attention matrices together. This fused result is then fed into a feedforward neural network to obtain the fused category context sequence F. The detailed implementation steps of MDF are illustrated in (4) to (7). It's important to emphasize that MDF, as a versatile module, has the capability to fuse various context information sequences as per the requirements of the application scenario, thus offering the potential for BMDF-SR to be applicable in diverse contexts.

We input the sequences into the multi-head attention mechanism for calculating attention scores, where h represents the number of attention heads, and d denotes the dimension of the item input sequence. The attention matrix for each sequence is computed using (4). This calculation allows the MDF module to effectively fuse the context sequences and learn their importance, enhancing the model's ability to make accurate predictions.

$$M_{attention}^{i} = (RW_Q^i)(RW_K^i) \tag{4}$$

Where $M_{attention}^{i} \in \mathbb{R}^{d*d}$ represents the attention matrix, $W_Q^i, W_K^i \in \mathbb{R}^{d*d}$ represents the query and key projection matrix, $R \in \mathbb{R}^{m*d}$ represents the input sequence, m represents the length of the input sequence, and i represents one branch of the multi-head attention mechanism. Then, according to (5), we obtain the attention output for each head.

$$M_{head}^{i} = softmax(\frac{M_{attention}^{i}}{\sqrt{d}})(RW_V^i) \tag{5}$$

Where $M^i_{head} \in \mathbb{R}^{d*d}$ represents the attention output for attention head i, and W^i_V represents the value projection matrix. Then, for each attention output, we perform concatenation according to (6) to obtain the final attention matrix for the input sequence.

$$A^S_{multihead} = concact(M^1_{head}, M^2_{head}, \ldots, M^h_{head})W^O \tag{6}$$

Where $A^S_{multihead} \in \mathbb{R}^{(d*h)*(d*h)}$ represents the attention scores for sequence S, and $W^O \in \mathbb{R}^{(d*h)*(d*h)}$ represents a learnable equal weight matrix. Following the above-mentioned approach, we obtain the attention matrix $A^{S_u}_{multihead}$ and $A^{S_c}_{multihead}$ for the hidden vector sequence $H^{(1)}$ and the category context embedding sequence S^e_c. Next, we fuse the attention matrices from different source inputs using (7) to obtain the attention matrix $A \in \mathbb{R}^{d*d*m}$.

$$A = A^{S_u}_{multihead} \otimes A^{S_c}_{multihead} \tag{7}$$

After performing the aforementioned steps, we achieve the decoupled fusion of attention calculation for multiple sequence inputs. Finally, we pass the attention matrix through a feed-forward layer to obtain the hidden vector sequence $F = \{f_1, f_2, \ldots, f_m\}$, which represents the fused category context sequence.

4.3 Semantic balance module

During our research, we discovered that the time context representation refers to the contextual relationships formed between each item in the time sequence and its preceding and succeeding items. On the other hand, the item sequence representation represents the user's interaction records over a specific period of time. Additionally, in our model, the item sequence incorporates other context sequences. Gong et al. (2023) Unlike NMT, there is no direct correspondence between these two types of sequences in our model, which results in a semantic asymmetry issue during the forward translation process from the item sequence S_u to the time context sequence S_t. To address this issue, we introduced a SBM to balance the semantic relationship between the two sequences, thereby achieving forward dependency modelling from the project sequence to the time context sequence. The SBM mainly consists of a Variational Autoencoder (VAE). The reason for choosing the VAE lies in its capacity as a generative model that can effectively learn latent representations within sequence data, thus generating sequences that are semantically similar to the target sequence. Li et al. (2023) The structure of the SBM module is illustrated in Fig. 3.

We feed the hidden vector sequence F into a VAE (Li et al., 2023) to compute the forward transformation vector z_k. This vector follows a normal distribution and is conditioned on the previous actions of the sequence.

$$q(z_k|f_k) = \mathcal{N}(\mu(f_k), diag\{\sigma^1(f_k)\}) \tag{8}$$

Where $\mu(f_k)$ and $\sigma^2(f_k)$ denote the standard normal distribution parameters [47] generated from f_k, respectively, and can be simply defined as (9).

$$\mu(f_k) = W^{(1)}_{f_k} + b_\mu, \sigma^2(f_k) = exp(W^{(2)}_{f_k} + b_\sigma) \tag{9}$$

Where $W^{(1)} \in \mathbb{R}^{(d/2*1)}$ and $W^{(2)} \in \mathbb{R}^{(d/2*1)}$ are two weight matrices that can be learned, the b_μ and b_σ are balance factors, respectively. By sampling the potential factor z_k from the above posterior probability distribution and using a reparameterization technique to avoid possible non-differentiation problems during training, the representation of item i is enhanced from a fixed-length vector f_k to a variable-length vector z_k by such a process, which in turn allows the semantics of the item sequence to be extended accordingly.

4.4 Three-layers seq2seq module

We feed the embedded sequence of item sequences $S_u^e = \{i_1^e, i_2^e, i_3^e, \ldots, i_m^e\}$ generated by user u in the past time into the first layer of the seq2seq neural network and output the hidden state $h_k^{(1)} \in \mathbb{R}^{(d*1)}$ of each item i_k^e at each time step k^{th} according to (10).

$$h_k^{(1)} = RNN(h_{h-1}^{(1)}, i_k^e) \tag{10}$$

The sequence of items S_u^e is obtained as a sequence of hidden vectors $H^{(1)} = \{h_1^{(1)}, h_2^{(1)}, h_3^{(1)}, \ldots, h_m^{(1)}\}$ after the first layer of the seq2seq network, in addition, as (11) uses $h_k^{(1)}$ in the loss function of the first layer of the recurrent neural network to maintain its ability to predict the next item.

$$\mathcal{L}^{(1)} = -log(p(i_{k+1}|h_k^{(1)})) + \mathbb{D}(q(h_k^{(1)}|x)\|p(h_k^{(1)})) \tag{11}$$

Then, we input it into the MDF module to obtain the hidden vector sequence F that fuses multiple sequence features into the SBM module to expand the semantics of the item sequences to obtain the forward transformation vector sequence $Z = \{z_1, z_2, z_3, \ldots, z_m\}$. Next, we input the forward transformation sequence Z that achieves the semantic balance and fusion of multiple sequence features into the second layer of the seq2seq neural network together with the temporal context embedding sequence S_t^e. The hidden state $h_k^{(2)} \in \mathbb{R}^{(d*1)}$ with a forward dependence from the item sequence S_u to the temporal context sequence S_t is obtained after (12).

$$h_k^{(2)} = RNN(h_{k-1}^{(2)}, t_k^e, W^{(2)} z_k) \tag{12}$$

Where $W^{(3)} \in \mathbb{R}^{(d*d/2*1)}$ is a learnable equal weight matrix, d denotes the dimensionality of the matrix. Then, as in (13) we modify the standard VAE loss function acting on the translation between the sequence of items to the sequence of contexts in the second layer.

$$\mathcal{L}^{(2)} = \mathcal{L}_t^{(2)} + KL_t^{(2)} \tag{13}$$

Where $KL_t^{(2)} = \mathbb{D}(q(z_k|x)\|p(z_k))$ is the relative entropy scattering regularization term from the VAE loss function and $\mathcal{L}_t^{(2)} = -log(p(t_k^e|h_k^{(2)}))$ is used to predict the next context of the item. After the second layer, we obtain the hidden vector sequence $H^{(2)} = \{h_1^{(2)}, h_2^{(2)}, \ldots, h_m^{(2)}\}$ with positive dependence from the item sequence S_u to the temporal context sequence S_t. In order to keep consistent with the elemental input, we input the initial item embedding sequence S_u^e with the hidden vector sequence $H^{(2)}$ into the third layer of the recurrent neural network, and obtain the hidden state vector $h_k^{(3)} \in \mathbb{R}^{(d*1)}$ with bidirectional dependence after (14).

$$h_k^{(3)} = RNN(h_{k-1}^{(3)}, i_k^e, h_k^{(2)}) \tag{14}$$

After the third layer of seq2seq neural network, we obtained the sequence of hidden vectors $H^{(3)} = \{h_1^{(3)}, h_2^{(3)}, \ldots, h_m^{(3)}\}$. Similarly, we adopted (15) as the loss function of the third layer of seq2seq, which uses the same principle as (10).

$$\mathcal{L}^{(3)} = -log(p(i_{k+1}|h_k^{(3)})) + \mathbb{D}(q(h_k^{(3)}|x)\|p(h_k^{(3)})) \tag{15}$$

4.5 Preference fusion module

We create a bidirectional dependency between the time context and item sequences using three stacked seq2seq modules, capturing dynamic user preference patterns. Recent research highlights the importance of including user static preferences for predicting the next item (Tang et al., 2021). Therefore, we initialise each user's most recent interactions as their static preference and have designed a PF module to integrate dynamic and static preferences. First, we fuse static preferences with dynamic preferences using a Kronecker product. Then, the user's final preference is generated using a VAE. The specific implementation steps are as follows:

For each user $u \in U$, a randomly initialized vector $S_{static} \in \mathbb{R}^{(d*1)}$ represents their general preferences. We obtain a hidden vector $h_k^{(3)} \in \mathbb{R}^{(d*1)}$ from $H^{(3)}$, denoting dynamic preferences. Then, we combine static and dynamic preferences to obtain the user's fused preferences $h_k \in R^{(d*1)}$, using (16).

$$h_k = h_k^{(3)} \bigotimes S_{static} \tag{16}$$

In the fusion process, we simply fuse the static and dynamic preferences at the end, ignoring the effect of the dynamic preference $h_k^{(3)}$ and easily back-propagating the gradient. Therefore we again engage the VAE module to generate the final representation $\widehat{y}$ from the approximate posterior probability $q(\widehat{y}|h_k)$, as shown in (17).

$$q(\widehat{y}|h_k) = \mathcal{N}(\mu(h_k), diag\{\sigma^2(h_k)\}) \tag{17}$$

Where $\mu(h_k)$ and $\sigma^2(h_k)$ are generated from h_k against the standard normal distribution, as shown in (18).

$$\mu(h_k) = W^{(3)} + b_\mu, \sigma^2(h_k) = exp(W^{(4)} + b_\sigma) \tag{18}$$

Where $W^{(3)} \in \mathbb{R}^{(d*2d)}$ and $W^{(4)} \in \mathbb{R}^{(d*2d)}$ are two weight matrices that can be trained. Finally, as in (19) we obtain the loss of the FP module by modifying the standard VAE loss.

$$\mathcal{L}_{PF} = \mathcal{L}_{(f)} + KL_{(f)} \tag{19}$$

Where $\mathcal{L}_{(f)} = -log(p(h_k|\widehat{y}))$ is introduced to consider the static interest of the user and $KL_{(f)} = \mathbb{D}(q(\widehat{y}|h_k)\|p(\widehat{y}))$ is the regularization term.

4.6 Optimize

There are five loss functions $\mathcal{L}_{SBM}$, $\mathcal{L}^{(1)}$, $\mathcal{L}^{(2)}$, $\mathcal{L}^{(3)}$, $\mathcal{L}_{PF}$, and $\mathcal{L}_{MDF}$ in our model, so we define the loss function of the BMDF-SR model as (20).

$$\mathcal{L}_{BMDF-SR} = \mathcal{L}_{SBM}+\mathcal{L}^{(1)}+\mathcal{L}^{(2)}+\mathcal{L}^{(3)}+\mathcal{L}_{PF}+\mathcal{L}_{MDF}+\lambda(KL_c^{(2)}+KL_i^{(2)}+KL_t^{(2)}) \tag{20}$$

Where λ is a hyperparameter$KL_c^{(2)}$, $KL_i^{(2)}$ and $KL_t^{(2)}$ is a regularization term, and $\lambda(KL_c^{(2)} + KL_i^{(2)} + KL_t^{(2)})$ will gradually converge to zero in our training, so its effect can be ignored. To optimize all the model parameters including the trainable weight matrix, we use the widely used Adam optimizer (Sun et al., 2021).

In addition, to reduce the risk of overfitting to noise in the training data and to prevent excessive dependence on certain neurons, thereby enhancing the model's generalization to unseen data, we introduced a dropout layer and set its rate to 0.2. Tang et al. (2021)

Table 1 Statistical information on datasets

Datasets	*Samples*	*Sparsity*	*l*	*m*
MovieLens-1M	999,611	0.9516	5	0
Gowalla	896,506	0.9971	10	20
JDshop	1,151,117	0.9983	5	20
Taobao	2,437,886	0.9985	20	0

5 Experiments

5.1 Database and evaluation metrics

Our experiments will be conducted on four public datasets:

- **MovieLens-1M** is one of the most classic datasets in the field of recommender systems. It is widely used for a variety of recommendation tasks and contains 1,000,209 rating records for 3,900 movies in 18 categories from 6,040 users, with each record associated with a specific timestamp.
- **Gowalla** is a public location-based social network dataset for point-of-interest recommendation tasks collected between February 2009 and October 2010.Each record in Gowalla consists of a user ID, a point-of-interest ID, a corresponding GPS location, and a timestamp.
- **Taobao** and **JDshop** are two large datasets on e-commerce from Taobao and JD, the two largest e-commerce platforms in China. Each record in these two datasets consists of a timestamp, a category ID, an item ID and a user ID. Other information such as brands and stores is also provided in the JDshop dataset.

During the data preprocessing phase, we removed records with user interaction counts of less than '*l*' and records with item user participation counts of less than '*l*'. We retained some users in Gowalla, MovieLens-1M, JDshop, and Taobao datasets with record counts exceeding 20, as previous deletions might have caused short interaction sequences. To address this, we used the item sequence's category change sequence as a category context sequence, which was fused with the item sequence in the MDF module for additional supervision signals. Additionally, we used the item sequence interactions' time information to create the time context sequence, described as a latent transition pattern in the model's second part. We selected the last user interaction sequence item for testing, the second-to-last for validation, and the others for training. To prevent excessively large embedding matrices and model misinterpretation, we limited interaction sequence length to the dataset's top 5%, ensuring its validity and authenticity. The preprocessed data for the four datasets are presented in Table 1. In addition, the hardware environment we used in our experiments is shown in Table 2.

Table 2 Information of the machine

Operating System	Ubuntu 18.04.5 LTS
CPU	Intel(R) Xeon(R) CPU E5-2640 v4
GPU	Tesla V100
GPU Memory	16GB
Memory	256GB

For evaluation metrics, we employ normalized discounted cumulative gain (NDCG@K), hit rate (HIT@K), and F-metric (F1@K). To ensure a comprehensive evaluation, we set K to 5 and 10, measuring the model's performance based on NDCG, HIT, and F1.

5.2 Baseline methods

We selected several serially recommended methods that have been widely used in the last five years as baseline methods:

- SASRec (Kang & McAuley, 2018) is an RNN-based sequential recommendation model that uses a dual network to model a set of targets and their corresponding interaction sequences, which is the first approach that divides the conversation into target and auxiliary sequences.
- BERT4Rec (Sun et al., 2019) is an RNN-based sequence recommendation model that uses a dual network to model a set of targets and their corresponding interaction sequences. This is the first approach that divides the conversation into target and auxiliary sequences.
- CBS (Le et al., 2018) is a sequential recommendation model that uses a dual network to model a set of goals and their corresponding interaction sequences. Notably, it is the first approach that divides the conversation into target and auxiliary sequences.
- STAR (Rakkappan & Rajan, 2019) is a multi-sequence sequential recommendation model that incorporates contextual information. It exploits the dependencies between item sequences and temporal and interval contexts by stacking multiple RNN structures to achieve high performance sequential recommendations.
- ATRank (Zhou et al., 2018) is a generic architecture designed to achieve a comprehensive user representation by widely fusing heterogeneous user behaviors. It employs attention mechanisms to model diverse user behaviors and incorporates behavioral information through behavior grouping and time interval encoding.
- CORE (Hou et al., 2022) is a session-based sequential recommendation method emphasizing representational consistency based on users' short-term behaviors within a session. It ensures that both sessions and items exist in the same representation space.
- S3Rec (Zhou et al., 2019) is a self-supervised learning framework for sequential recommendations, utilizing multiple information sources such as user behavior sequences and item attributes to learn item representations. By leveraging mutual information maximization, it fully exploits the connections among the inputs.

To effectively experiment with the baseline methods used in our experiments, we employed the RecBole (Zhao et al., 2021) library, an open-source repository for RS. This library offers access to various experimental datasets and enables efficient implementation of baseline experiments in a consistent environment. To maintain consistency with the baseline comparison experiments, we kept the latent state vector and embedding vector dimensions at 64 and did not alter other parameters. We added a dropout layer to the model and set it to 0.2 to reduce overfitting during training. For model training, we segmented the training sequences using a sliding window with a length of 5, setting the window size to 550 for the MovieLens-1M dataset, 200 for Gowalla, 130 for the JDshop dataset, and 120 for the Taobao dataset.Additionally, we set the batch size to 128 and the learning rate to 0.001. Hyperparameters were set to 1 for MovieLens-1M, 20 for Gowalla and JDshop, and 40 for Taobao. We also employed an annealing strategy to optimize performance.

Table 3 Comparison of experimental results

		SASRec	BERT4Rec	CBS	STAR	ATRank	CORE	S3Rec	BMDF-SR
MovieLens-1M	HIT@5	0.4421	0.4430	0.4577	0.3803	0.3571	0.4544	0.4577	**0.5149**
	HIT@10	0.5829	0.5762	0.5749	0.4891	0.3856	0.5891	0.5866	**0.6349**
	NDCG@5	0.3066	0.3159	0.3376	0.2729	0.1831	0.3298	0.3321	**0.3873**
	NDCG@10	0.3522	0.3601	0.3763	0.3082	0.2240	0.3621	0.3688	**0.4223**
	F1@5	0.1473	0.1474	0.1526	0.1268	0.1178	0.1529	0.1489	**0.1713**
	F1@10	0.1065	0.1039	0.1058	0.0889	0.0718	0.1057	0.1065	**0.1153**
Gowalla	HIT@5	0.7033	0.6881	0.6676	0.4927	0.6871	0.7032	0.7056	**0.7627**
	HIT@10	0.8027	0.7796	0.7659	0.5992	0.7679	0.8066	0.8107	**0.8434**
	NDCG@5	0.5729	0.5763	0.5481	0.3871	0.4571	0.5753	0.5755	**0.6545**
	NDCG@10	0.6021	0.6067	0.5796	0.4215	0.5215	0.6049	0.6067	**0.6886**
	F1@5	0.2345	0.2293	0.2225	0.1642	0.2236	0.2368	0.2357	**0.2576**
	F1@10	0.1459	0.1421	0.1389	0.1091	0.1401	0.1498	0.1463	**0.1551**
JDshop	HIT@5	0.6794	0.6695	0.6625	0.5177	0.5644	0.6824	0.6812	**0.7648**
	HIT@10	0.7556	0.7311	0.7254	0.5874	0.6125	0.7616	0.7611	**0.8185**
	NDCG@5	0.5882	0.5940	0.5873	0.4458	0.4551	0.5889	0.5922	**0.6792**
	NDCG@10	0.6128	0.6141	0.6078	0.4681	0.4998	0.6133	0.6131	**0.6981**
	F1@5	0.2266	0.2231	0.2208	0.1727	0.2197	0.2294	0.2331	**0.2554**
	F1@10	0.1375	0.1329	0.1319	0.1068	0.1311	0.1381	0.1388	**0.1493**
Taobao	HIT@5	0.4152	0.4421	0.4408	0.3054	0.3329	0.4432	0.4166	**0.5151**
	HIT@10	0.5146	0.5275	0.5239	0.3701	0.4356	0.5321	0.5138	**0.6017**
	NDCG@5	0.3218	0.3552	0.3548	0.2446	0.3201	0.3652	0.3328	**0.4206**
	NDCG@10	0.3538	0.3828	0.3817	0.2655	0.3529	0.3964	0.3638	**0.4469**
	F1@5	0.1384	0.1474	0.1469	0.1569	0.1581	0.1468	0.1421	**0.1713**
	F1@10	0.0935	0.0961	0.0953	0.1021	0.1055	0.0971	0.0955	**0.1102**

Optimal and suboptimal performance values are indicated with bold and underlined text, respectively

5.3 Contrast experiment

We carried out comparative experiments on four datasets to evaluate the performance of the BMDF-SR method against the baseline methods using the metrics NDCG@K, HIT@K, and F1@K, where K is set to 5 and 10. The experimental results are presented in Table 3 and Fig. 4. The results demonstrate that the BMDF-SR method significantly outperforms all the baseline methods across the four datasets.

In addition, the CBS and STAR methods in the comparison experiments are both sequential recommendation methods with multiple sequences, and they show suboptimal performance on the datasets MovieLens-1M and Taobao. However, they do not dominate compared with S3Rec and CORE methods on the four datasets. There are two main reasons for this phenomenon: First, they only consider the one-way dependence of item sequences on auxiliary sequences. Second, they ignore the semantic asymmetry problem that exists between different source sequences. Finally, among the non-multi-sequence approaches, SASRec and BERT4Rec achieved excellent performance by exploiting the properties of the attention mechanism. However, BERT4Rec outperforms SASRec on four datasets by establishing bidirectional dependencies on item sequences. this result demonstrates in another dimension that establishing bidirectional dependencies can improve model performance.

5.4 Ablation study

We designed ablation experiments for the SBM module, PF module and MDF module to examine the effects of semantic balancing, decoupling fusion of multiple sequences and preference fusion on model performance, and these ablation studies can be divided into:

- -SBM: indicates that the SBM is excluded.
- -PF: indicates that the PF module is excluded.
- -V: indicates that the SBM and PF module are excluded.
- -MDF: indicates that the MDF module is excluded, and the model will only model the bi-directional dependencies between item sequences and temporal contexts.
- -ALL: indicates that the SBM, MDF, and PF module are excluded.

The ablation experiments will compare the performance on four evaluation metrics HIT@K, NDCG@K, and F1@K, where the values of K are taken as 5 and 10. The experimental results are shown in Table 4.

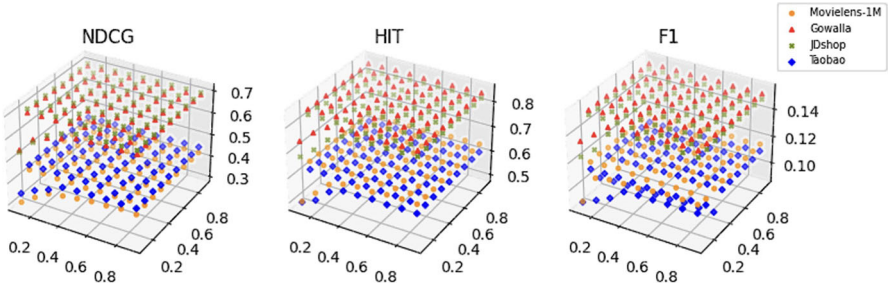

Fig. 4 Scatter plot of experimental results of BMDF-SR on four data sets. BMDF-SR shows rapid stabilization after rapid rise on three evaluation indicators, which indicates that BMDF-SR is trainable

Table 4 The Ablation Experiment On Four Datasets

		-SBM	-PF	-V	-MDF	-ALL	BMDF-SR
MovieLens-1M	HIT@5	0.5118	0.5078	0.4978	0.5060	0.4822	**0.5149**
	HIT@10	0.6316	0.6207	0.6017	0.6318	0.5927	**0.6349**
	NDCG@5	0.3721	0.3721	0.3689	0.3820	0.3583	**0.3873**
	NDCG@10	0.4108	0.4087	0.4036	0.4030	0.3919	**0.4223**
	F1@5	0.1656	0.1621	0.1602	0.1653	0.1545	**0.1713**
	F1@10	0.1101	0.1078	0.1055	0.1094	0.0998	**0.1153**
Gowalla	HIT@5	0.7579	0.7541	0.7501	0.7576	0.7439	**0.7627**
	HIT@10	0.8401	0.8392	0.8362	0.8392	0.8276	**0.8434**
	NDCG@5	0.6499	0.6428	0.6398	0.6518	0.6297	**0.6545**
	NDCG@10	0.6781	0.6726	0.6698	0.6779	0.6658	**0.6886**
	F1@5	0.2526	0.2477	0.2468	0.2525	0.2421	**0.2576**
	F1@10	0.1522	0.1520	0.1520	0.1526	0.1488	**0.1551**
JDshop	HIT@5	0.7456	0.7431	0.7348	0.7461	0.7267	**0.7648**
	HIT@10	0.7952	0.7896	0.7795	0.7876	0.7701	**0.8185**
	NDCG@5	0.6671	0.6478	0.6478	0.6688	0.6389	**0.6792**
	NDCG@10	0.6742	0.6711	0.6603	0.6824	0.6524	**0.6981**
	F1@5	0.2455	0.2356	0.2275	0.2421	0.2188	**0.2554**
	F1@10	0.1425	0.1387	0.1291	0.1426	0.1201	**0.1493**
Taobao	HIT@5	0.5121	0.5074	0.5017	0.5112	0.4989	**0.5151**
	HIT@10	0.5919	0.5892	0.5872	0.5928	0.5796	**0.6017**
	NDCG@5	0.4169	0.4179	0.4146	0.4128	0.4127	**0.4206**
	NDCG@10	0.4428	0.4452	0.4307	0.4438	0.4289	**0.4469**
	F1@5	0.1641	0.1656	0.1501	0.1632	0.1581	**0.1713**
	F1@10	0.1065	0.1055	0.0998	0.1050	0.0967	**0.1102**

The best results of the model are shown in bold, and the second-best result is underlined

The results of the six groups of ablation experiments show a clear reduction in performance when core components such as the SBM or MDF modules are omitted, or when changes are made to other essential modules within the BMDF-SR framework. Firstly, when comparing the -ALL and -MDF models, it's clear that the experimental results for -MDF exceed those for -ALL. However, both models show a significant drop in performance over the four data sets. However, they still outperform the mainstream SR models mentioned in the comparative experiments, highlighting the effectiveness of the three-layer seq2seq structure of BMDF-SR and the MDF module in SR. Secondly, comparing the -V, -PF, and -SBM models, it is evident that the -V and -PF models show a noticeable drop in performance, especially on the sparsely populated MovieLens-1M dataset. However, the model that retains the PF module and removes the SBM module does not show a significant drop in performance, which suggests that the PF module has a particular advantage in dealing with the common problem of data sparsity in RS. Finally, when we compare the performance of -SBM with the full BMDF-SR model, we observe that the performance degradation is less pronounced on the simple MovieLens-1M dataset. However, as the complexity of the dataset increases, especially in

Table 5 Experimental results of hyperparameters for the number of sliding windows L

L	Gowalla			JDshop		
	HIT@10	F1@10	NDCG@10	HIT@10	F1@10	NDCG@10
2	0.7655	0.1406	0.5909	0.7647	0.1390	0.6358
3	0.7975	0.1459	0.6255	0.7713	0.1403	0.6434
4	0.8027	0.1515	0.6431	0.7889	0.1435	0.6676
5	0.8434	**0.1551**	**0.6786**	0.8185	**0.1493**	**0.6981**
6	**0.8441**	0.1549	0.6771	0.8185	0.1451	0.6933
7	0.8301	0.1529	0.6743	**0.8196**	0.1425	0.6933
8	0.8293	0.1505	0.6664	0.8050	0.1392	0.6837
9	0.8148	0.1488	0.6598	0.7923	0.1356	0.6721
10	0.8041	0.1421	0.6445	0.7872	0.1298	0.6673

The best results of the model are shown in bold, and the second-best result is underlined

challenging scenarios, the performance drop for -SBM becomes more noticeable, which suggests that the SBM is better suited to handle complex situations.

5.5 Parametric experiment

We designed multiple sets of experiments to test the complexity of the model and find the optimal parameters based on the characteristics of the BMDF-SR model. Due to the limited length of the article, we only show the results of the parametric experiments on the Gowalla and JDshop datasets.

Firstly, to scrutinize the impact of hyperparameters on the model, we executed experiments concerning the number of sliding windows, L. This is crucial as excessive sliding windows might exacerbate the dataset's sparsity issue, thereby affecting the model's training efficacy and recommendation performance. Conversely, too few sliding windows could hinder the model from effectively capturing valuable information embedded in lengthier user behavior sequences, consequently diminishing recommendation performance. As a result, we devised multiple experimental sets across four datasets to gauge the influence of the hyperparameter L on the BMDF-SR model's performance. We varied the L value between 2 and 10, and assessed the findings using three evaluation metrics (HIT@10, NDCG@10, and F1@10). As shown in Table 5 and Fig. 5(a, b), the performance of the model is relatively balanced when the number of sliding windows is set to 5.

Next, we established multiple hyperparameter experiments for the sliding window size. On the one hand, a smaller sliding window size may hinder the model from capturing the long-term user interest evolution and could disrupt the integrity of the sequence pattern. On the other hand, a larger sliding window size may make it difficult for the model to extract useful information from the long-term user behavior sequence, potentially introducing additional noise and resulting in performance degradation. We set the sliding window size to take a range of values from 60 to 600, with an increment of 10 and assessed the results using four evaluation metrics (HIT@K, NDCG@K, F1@K, and AUC). The experimental results are shown in Table 6 and Fig. 5(c, d), the sliding window size should be set to 550 on the

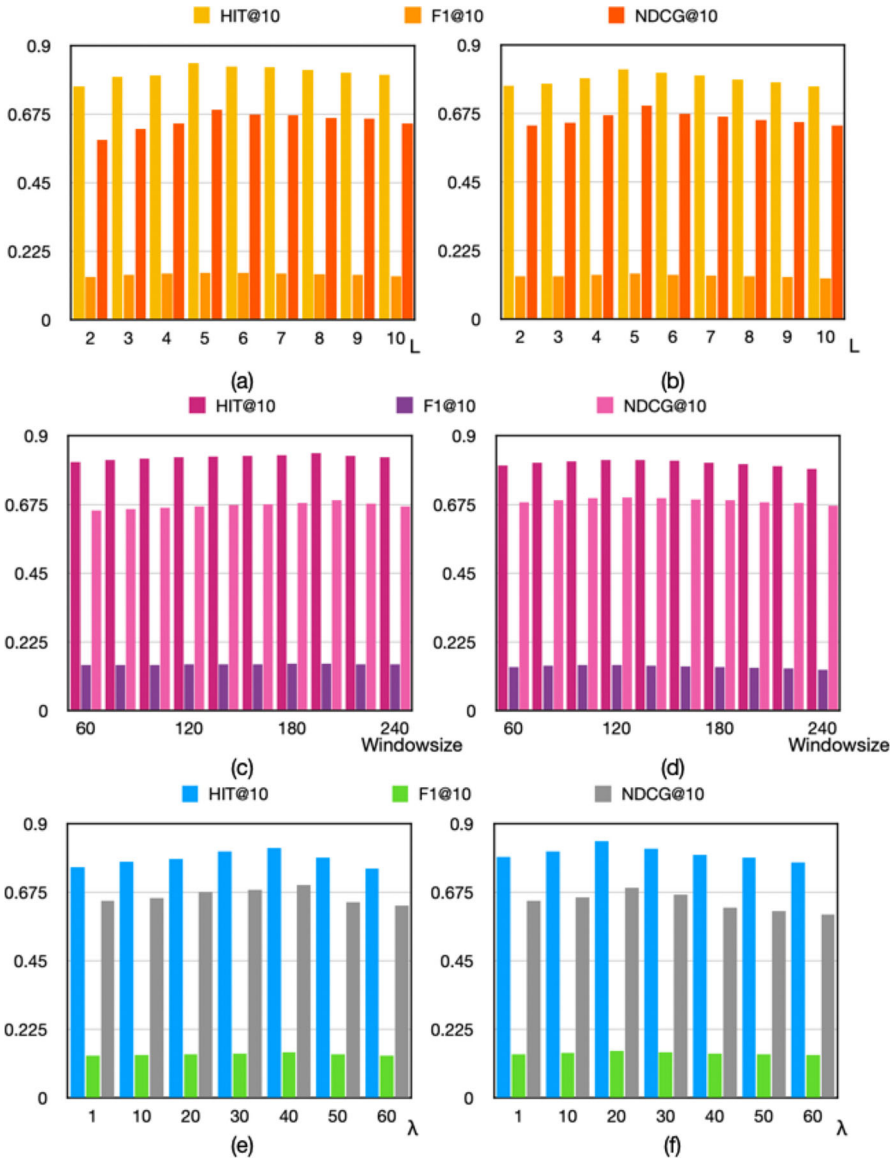

Fig. 5 Results of three sets of parametric experiments. Where (a), (c) and (e) are the results of experiments on the Gowalla dataset and (b), (d) and (f) are the results of experiments on the JDshop dataset

MovieLens-1M dataset, 200 on the Gowalla dataset, 120 on the JDshop dataset, and 120 on the Taobao dataset.

Finally, we devised multiple experimental sets to assess the impact of the balance factor λ on the BMDF-SR model's performance. We set the balancing factor λ to "1, 10, 20, 30, 40, 50, 60" and carried out experiments on four datasets, with the results displayed in Table 7

Table 6 Hyperparametric experimental results for sliding window size

WindowSize	Gowalla			JDshop		
	HIT@10	F1@10	NDCG@10	HIT@10	F1@10	NDCG@10
60	0.8131	0.1478	0.6555	0.8021	0.1411	0.6824
80	0.8186	0.1487	0.6596	0.8098	0.1453	0.6887
100	0.8233	0.1496	0.6635	0.8136	0.1479	0.6956
120	0.8288	0.1505	0.6687	0.8185	**0.1493**	**0.6981**
140	0.8313	0.1515	0.6721	**0.8187**	0.1468	0.6948
160	0.8321	0.1516	0.6754	0.8168	0.1448	0.6901
180	0.8356	0.1518	0.6785	0.8105	0.1422	0.6887
200	**0.8434**	**0.1521**	**0.6886**	0.8057	0.1398	0.6824
220	0.8432	0.1518	**0.6885**	0.7978	0.1366	0.6796
240	0.8287	0.1501	0.6787	0.7901	0.1321	0.6716

The best results of the model are shown in bold, and the second-best result is underlined

and Fig. 5(e, f). The λ has minimal influence on the MovieLens-1M dataset, and the best performance is achieved when $\lambda = 1$. This is because the dataset has fewer categories, and semantic balancing is not necessary for datasets with fewer categories. However, significant semantic imbalance is observed in the Taobao, JDshop, and Gowalla datasets, making semantic balancing a crucial component for these datasets. Furthermore, we discovered that the regularization parameter results in better overall performance when encountering relatively large values of λ, thereby preventing overfitting. Consequently, we choose λ as 20 for the Gowalla and Taobao datasets and λ as 40 for the JDshop dataset.

5.6 Complexity study

We also explored the training efficiency of the BMDF-SR model by analyzing the learning rate of seq2seq and MDF modules at each layer across four differen We illustrated the loss variation curves for these modules on the four datasets, as displayed in Fig. 6. The relative

Table 7 Experimental results for the equilibrium factor λ

λ	Gowalla			JDshop		
	HIT@10	F1@10	NDCG@10	HIT@10	F1@10	NDCG@10
1	0.7921	0.1429	0.6476	0.7571	0.1379	0.6474
10	0.8096	0.1472	0.6588	0.7747	0.1408	0.6560
20	0.8434	**0.1551**	**0.6886**	0.7844	0.1428	0.6771
30	**0.8439**	0.1488	0.6836	**0.8188**	0.1455	0.6842
40	0.8322	0.1445	0.6239	0.8146	**0.1493**	**0.6981**
50	0.8189	0.1432	0.6125	0.7890	0.1435	0.6432
60	0.8021	0.1407	0.6021	0.7522	0.1377	0.6321

The best results of the model are shown in bold, and the second-best result is underlined

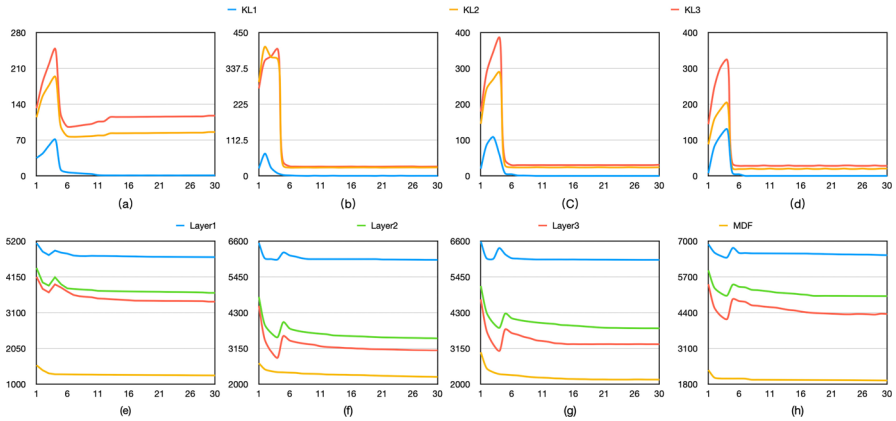

Fig. 6 The variation curves of loss, where (a), (b), (c) and (d) indicate the variation of KL loss on MovieLens-1M, Gowalla, JDshop and Taobao datasets, respectively; (e), (f), (g) and (h) indicate the variation of loss per layer of seq2seq and MDF modules on MovieLens-1M, Gowalla, JDshop and Taobao datasets, respectively.Where X-axis denotes epoch and Y-axis denotes the calculated value of loss

entropy loss of each layer can be observed to decrease rapidly and converge following a brief increase, indicating that the model is easy to train.

In addition, we compared the computational cost with the mainstream 5 baseline models, and the comparison results are shown in Table 8. In this experiment, we used the same equipment as in Table 2 for the experiment to ensure consistency of the experimental environment. The experimental results show that the CBS, SASRec, and CORE methods have similar training efficiency, while the S3Rec and STAR methods have significantly increased in computational cost. In general, BMDF-SR achieves better performance than CBS, SASRec, and CORE methods without a significant increase in computational cost, while BMDF-SR achieves better performance and a reduction in computational cost compared to S3Rec and STAR methods.

Table 8 Time consumption statistics for the main baseline model on the Gowalla dataset

Methods	Time-Consuming/Epoch	Converge Epoch	Total Time-Consuming
SASRec	87.48s	100	8748.0s
CBS	99.35s	100	9935.0s
CORE	52.21s	200	10442.0s
S3Rec	6636.90s	10	66369.0s
STAR	1389.27s	20	27785.4s
BMDF-SR	253.1s	40	10123.0s

When the evaluation metrics are relatively stable, we obtain the converged Epoch by rounding

6 Conclusion

We propose a bidirectional multi-sequence decoupling fusion model for sequential recommendation. (BMDF-SR) Unlike traditional SR models that simply merge context information with item sequences during the embedding phase, our approach treats context information as independent sequences and computes attention matrices for each of them separately, which effectively reduces the complexity of attention matrix computation. Additionally, the attention matrices for each context sequence are independently calculated, reducing interference between heterogeneous information when merging attention matrices, which enables the model to distinguish between these diverse elements of information effectively. Furthermore, we adopt a stacked three-layer seq2seq approach to model the bidirectional dependency relationship between item sequences and context sequences, thereby enhancing the model's robustness to noise and missing data in the sequences. Moreover, to address the cold-start issue, we design a preference fusion module (PF) to integrate users' static preferences, thereby improving the model's accuracy in predicting the next interaction. We conducted extensive experiments on four real datasets, demonstrating the superiority of our proposed BMDF-SR method over baseline approaches.

Acknowledgements Thanks to Professor Qin Jiwei for his help and support during the research process.

Author Contributions -Aohua Gao: Writing, Main idea, Experiments, Analysis
-Jiwei Qin: Provide guidance
-Chao Ma and Tao Wang: Analysis

Funding This work was supported by the Science Fund for Outstanding Youth of Xinjiang Uygur Autonomous Region under Grant No. 2021D01E14.

Data Availability -MovieLens-1M: Availabile at https://grouplens.org/datasets/movielens/1m/.
-Gowalla: Availabile at https://snap.stanford.edu/data/loc-gowalla.html/.
-JDshop: Availabile at https://www.jd.com/.
-Taobao : Available at https://tianchi.aliyun.com/dataset/dataDetail?dataId=649/.

Code Availability The code will be released at https://github.com/WayneHuahua/BMDF-SR.git.

Declarations

Ethical Approval Not applicable as this study did not involve human participants.

Consent for Publication The authors grant the Publisher an exclusive licence to publish the article

Competing interests The authors declare no competing interests.

References

Dang, Y., Yang, E., & Guo, G., et al. (2023). Uniform sequence better: Time interval aware data augmentation for sequential recommendation. In: *Proceedings of the AAAI conference on artificial intelligence*. AAAI, Washington DC, USA. https://doi.org/10.1609/aaai.v37i4.25540

Duan, J., Zhang, P. F., Qiu, R., et al. (2023). Long short-term enhanced memory for sequential recommendation. *World Wide Web, 26*, 561–583. https://doi.org/10.1007/s11280-022-01056-9

Garcin, F., Dimitrakakis, C., & Faltings, B. (2013). Personalized news recommendation with context trees. In: *Proceedings of the 7th ACM conference on recommender systems*. ACM, Hong Kong, China. https://doi.org/10.1145/2507157.2507166

Gong, J., Wan, Y., Liu, Y., et al. (2023). Reinforced moocs concept recommendation in heterogeneous information networks. *ACM Transactions on the Web, 17*, 1–27. https://doi.org/10.1145/3580510

Guo, L., Zhang, J., Chen, T., et al. (2023). Reinforcement learning-enhanced shared-account cross-domain sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering, 35*, 7397–7411. https://doi.org/10.1109/TKDE.2022.3185101

Hao, Y., Ma, J., Zhao, P., et al. (2023). Multi-dimensional graph neural network for sequential recommendation. *Pattern Recognition, 139*, 109504. https://doi.org/10.1016/j.patcog.2023.109504

He, R., Kang, W.C., & McAuley, J. (2017). Translation-based recommendation. In: *Proceedings of the eleventh ACM conference on recommender systems*. ACM, Como, Italy. https://doi.org/10.1145/3109859.3109882

Hidasi, B., & Karatzoglou, A. (2018). Recurrent neural networks with top-k gains for session-based recommendations. In: *Proceedings of the 27th ACM international conference on information and knowledge management*. ACM, Torino, Italy. https://doi.org/10.1145/3269206.3271761

Hidasi, B., Karatzoglou, A., & Baltrunas, L., et al. (2015). Session-based recommendations with recurrent neural networks. arXiv preprint arXiv:1511.06939, https://doi.org/10.48550/arXiv.1511.06939

Hou, Y., Hu, B., & Zhang, Z., et al. (2022). Core: simple and effective session-based recommendation within consistent representation space. In: *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. ACM, Madrid, Spain. https://doi.org/10.1145/3477495.3531955

Kang, W.C., & McAuley, J. (2018). Self-attentive sequential recommendation. In: *2018 IEEE International conference on data mining (ICDM)*. IEEE, Singapore. https://doi.org/10.1109/ICDM.2018.00035

Le, D.T., Lauw, H.W., & Fang, Y. (2018) Modeling contemporaneous basket sequences with twin networks for next-item recommendation. In: *Proceedings of the twenty-seventh international joint conference on artificial intelligence*. IJCAI, Stockholm, Sweden. https://doi.org/10.24963/ijcai.2018/474

Lei, J., Li, Y., Yang, S., et al. (2022). Two-stage sequential recommendation for side information fusion and long-term and short-term preferences modeling. *Journal of Intelligent Information Systems, 59*, 657–677. https://doi.org/10.1007/s10844-022-00723-7

Li, J., Wang, Y., & McAuley, J. (2020). Time interval aware self-attention for sequential recommendation. In: *Proceedings of the 13th international conference on web search and data mining*. ACM, Houston, TX. https://doi.org/10.1145/3336191.3371786

Ling, Z. H., Ai, Y., Gu, Y., et al. (2018). Waveform modeling and generation using hierarchical recurrent neural networks for speech bandwidth extension. *IEEE/ACM Transactions on Audio, Speech, and Language Processing, 26*, 883–894. https://doi.org/10.1109/TASLP.2018.2798811

Li, P., Que, M., & Tuzhilin, A. (2023). Dual contrastive learning for efficient static feature representation in sequential recommendations. *IEEE Transactions on Knowledge and Data Engineering, 1*, 1–13. https://doi.org/10.1109/TKDE.2023.3289469

Liu, C., Li, X., & Cai, G., et al. (2021). Noninvasive self-attention for side information fusion in sequential recommendation. In: *Proceedings of the AAAI conference on artificial intelligence*. AAAI, Palo Alto, California. https://doi.org/10.1609/aaai.v35i5.16549

Liu, Q., Wu, S., & Wang, D., et al. (2016). Context-aware sequential recommendation. In: *2016 IEEE 16th International conference on data mining*. IEEE, Barcelona, Spain. https://doi.org/10.1109/ICDM.2016.0135

Li, L., Xiahou, J., Lin, F., et al. (2023). Distvae: distributed variational autoencoder for sequential recommendation. *Knowledge-Based Systems, 264*, 110313. https://doi.org/10.1016/j.knosys.2023.110313

Rakkappan, L., & Rajan, V. (2019). Context-aware sequential recommendations withstacked recurrent neural networks. In: *The world wide web conference*. ACM, San Francisco, CA. https://doi.org/10.1145/3308558.3313567

Rendle, S., Freudenthaler, C., & Schmidt-Thieme, L. (2010). Factorizing personalized markov chains for next-basket recommendation. In: *Proceedings of the 19th international conference on world wide web*. ACM, Raleigh, North Carolina. https://doi.org/10.1145/1772690.1772773

Ren, J., & Gan, M. (2023). Mining dynamic preferences from geographical and interactive correlations for next poi recommendation. *Knowledge and Information Systems, 65*, 183–206. https://doi.org/10.1007/s10115-022-01749-7

Sun, F., Liu, J., & Wu, J., 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*. ACM, Beijing, China. https://doi.org/10.1145/3357384.3357895

Sun, K., Qian, T., Chen, X., et al. (2021). Context-aware seq2seq translation model for sequential recommendation. *Information Sciences, 581*, 60–72. https://doi.org/10.1016/j.ins.2021.09.001

Tang, J., & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional sequence embedding. In: *Proceedings of the eleventh ACM international conference on web search and data mining*. ACM, Marina Del Rey, CA. https://doi.org/10.1145/3159652.3159656

Tang, H., Zhao, G., Bu, X., et al. (2021). Dynamic evolution of multi-graph based collaborative filtering for recommendation systems. *Knowledge-Based Systems, 228*, 107251. https://doi.org/10.1016/j.knosys.2021.107251

Vaswani, A., Shazeer, N., & Parmar, N., et al. (2017). Attention is all you need. In: *roceedings of the 31st international conference on neural information processing systems*. Curran Associates, Inc., Long Beach, California. https://doi.org/10.48550/arXiv.1706.03762

Wang, S., Hu, L., & Cao, L., et al. (2018). Attention-based transactional context embedding for next-item recommendation. In: *Proceedings of the AAAI conference on artificial intelligence*. AAAI, New Orleans, Lousiana. https://doi.org/10.1609/aaai.v32i1.11851

Wang, C., Ma, W., Chen, C., et al. (2023). Sequential recommendation with multiple contrast signals. *ACM Transactions on Information Systems, 41*, 1–27. https://doi.org/10.1145/3522673

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*. ACM, Madrid, Spain. https://doi.org/10.1145/3477495.3531963

Ye, X., & Liu, D. (2022). A cost-sensitive temporal-spatial three-way recommendation with multi-granularity decision. *Information Sciences, 589*, 670–689. https://doi.org/10.1016/j.ins.2021.12.105

Yuan, X., Duan, D., & Tong, L., et al. (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*. ACM, Virtual Event, Canada. https://doi.org/10.1145/3404835.3463060

Yuan, W., Wang, H., Yu, X., et al. (2020). Attention-based context-aware sequential recommendation model. *Information Sciences, 510*, 122–134. https://doi.org/10.1016/j.ins.2019.09.007

Zhang, T., Zhao, P., & Liu, Y., et al. (2019). Feature-level deeper self-attention network for sequential recommendation. In: *Proceedings of the twenty-eighth international joint conference on artificial intelligence*. IJCAI, Macao, China. https://doi.org/10.24963/ijcai.2019/600

Zhang, Y., Yang, B., Liu, H., et al. (2023). A time-aware self-attention based neural network model for sequential recommendation. *Applied Soft Computing, 133*, 109894. https://doi.org/10.1016/j.asoc.2022.109894

Zhao, W.X., Mu, S., & Hou, Y., 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*. ACM, Virtual Event, Queensland. https://doi.org/10.1145/3459637.3482016

Zhong, C., Xiong, F., Pan, S., et al. (2023). Hierarchical attention neural network for information cascade prediction. *Information Sciences, 622*, 1109–1127. https://doi.org/10.1016/j.ins.2022.11.163

Zhou, C., Bai, J., & Song, J., et al. (2018). Atrank: An attention-based user behavior modeling framework for recommendation. In: *Proceedings of the AAAI conference on artificial intelligence*. AAAI, New Orleans, Lousiana. https://doi.org/10.1609/aaai.v32i1.11618

Zhou, K., Wang, H., & Zhao, W.X., et al. (2019). S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In: *Proceedings of the 29th ACM international conference on information & knowledge management*. ACM, Virtual Event, Ireland. https://doi.org/10.1145/3340531.3411954

Zhou, W., Liu, Y., Li, M., et al. (2023). Dynamic multi-objective optimization framework with interactive evolution for sequential recommendation. *IEEE Transactions on Emerging Topics in Computational Intelligence, 7*, 1228–1241. https://doi.org/10.1109/TETCI.2023.3251352