Neural Computing and Applications (2024) 36:9707–9721
https://doi.org/10.1007/s00521-023-09044-4

S.I.: NEURAL NETWORKS AND MACHINE LEARNING EMPOWERED METHODS AND APPLICATIONS IN HEALTHCARE

Sequential recommendation based on multipair contrastive learning with informative augmentation

Pei Yin[1,2] · Jun Zhao[1] · Zi-jie Ma[1] · Xiao Tan[1]

Received: 18 May 2023 / Accepted: 6 September 2023 / Published online: 5 October 2023

Abstract

To solve the recommendation accuracy degradation problem encountered in sequential recommendation cases caused by data sparsity—such as short historical user behaviour sequences and limited information—this paper proposes a sequential recommendation model based on multipair contrastive learning with informative augmentation (IA-MPCL). The model aims to better learn user preference representations. Initially, a self-attention network is utilized to maintain the intrinsic relevance of the original sequences and introduce virtual interaction items for short sequences to achieve informative enhancement. Subsequently, multiple positive samples are generated by data augmentation methods to form multiple pairs of positive and negative samples. A multipair contrastive loss is constructed to eliminate the negative impact of fake positive and negative samples on the training process of the self-attention network. Finally, an adaptive loss weighting mechanism is proposed to dynamically regulate the role of the contrastive loss during multitask training. Through comparison experiments involving baseline methods and experiments conducted on datasets with different sparsity levels, the results show that IA-MPCL achieves significant improvements in terms of both recommendation accuracy and data sparsity resistance.

Keywords Sequential recommendation · Data sparsity · Self-attention network · Contrastive learning · Representation learning

1 Introduction

User interest modelling stands as a pivotal concern in the realm of recommendation system research, hinging on the foundation of user behaviours over time. Consequently, the spotlight has turned to recommendation systems rooted in historical user behaviour sequences, and these systems have become subjects of considerable attention in recent years. Sequential recommendation endeavours to distil item relevance from past behaviours, culminating in the acquisition of preference representations that mirror user inclinations. This endeavour typically involves methodologies such as Markov chains [1], models reliant on recurrent neural networks (RNNs) [2], and self-attention models based on transformers [3].

Nonetheless, the challenge of data sparsity poses a hurdle, making it complicated to accurately unravel user interests in instances where the given sequences are succinct and comprise a limited set of interaction behaviours. The dearth of information within these sequences integrates complexity into the process of precisely capturing user preferences. Additionally, user data remain susceptible to the intrusion of noise, a scenario wherein specific past user actions may not faithfully depict their authentic

✉ Pei Yin
pyin@usst.edu.cn

Jun Zhao
skyusst@163.com

Zi-jie Ma
mazijie_m@163.com

Xiao Tan
tanx2021@163.com

[1] Business School, University of Shanghai for Science and Technology, Shanghai 200093, China

[2] School of Intelligent Emergency Management, University of Shanghai for Science and Technology, Shanghai 200093, China

interests. Consider an examine involving an e-commerce platform, where a user might inadvertently click on a product link, inadvertently transmitting inaccurate click-through rate (CTR) data to the system.

To address these challenges, researchers have proposed sequential recommendation approaches that are grounded in self-supervised contrastive learning. This strategy harnesses data augmentation to establish positive samples while accentuating the consistency among these positive samples and differentiating them from negative samples. This approach facilitates the acquisition of high-quality user representations from unlabelled data. For instance, CL4SRec [4] introduces a trio of data augmentation methods (masking, cropping, and reordering) to forge augmented views. These augmented views originating from the same user are designated as positive samples, while those originating from distinct users are categorized as negative samples. Consequently, a contrastive learning task is employed to mitigate the adverse effects of data sparsity to a certain extent. Furthermore, during the creation of augmented views, the masking and cropping operations effectively filter out noisy items, thereby heightening the noise resilience of the sequential recommendation model. On a divergent note, Duorec [5] adopts an unsupervised dropout mechanism alongside the supervised positive sampling process to create positive samples. This hybrid approach aims to conserve the holistic semantic information woven within the input sequences.

Although the previously mentioned self-supervised contrastive learning methods have demonstrated significant advancements in user representation learning, they still possess certain limitations.

(a) They exhibit subpar performance in cases with short sequences, as the models struggle to acquire precise preference representations for users with brief sequences. This limitation results in improper recommendations. (b) The existing self-supervised contrastive learning frameworks utilize single-pair positive samples (two augmented views from the same user) for constructing their contrastive losses. In scenarios where the disparity between these two positive samples is minimal, the model encounters difficulty when extracting valuable information from them, leading to the creation of false-positive samples. Moreover, the generated negative samples (two augmented views from different users) may not strictly depict negative instances, as they might display considerable consistency. This consistency prevents the model from effectively capturing their differences, ultimately producing false-negative samples. The presence of such erroneous positive and negative samples directly impacts the efficiency of the model training process. (c) Current contrastive learning-based recommendation approaches assign a fixed weight to the contrastive loss during multitask training. This practice impedes model convergence and ultimately influences the effectiveness of training.

After the above analysis, this paper puts forwards the following three research questions.

1. How can the preferences of users with short and sparse sequences be effectively represented?
2. How can the contrastive learning method use positive and negative samples more effectively for model training?
3. How can the training efficiency of the utilized model be improved by dynamically adjusting the contrastive loss weight?

Regarding these challenges, this paper introduces a model named “sequential recommendation based on multipair contrastive learning with informative augmentation” (IA-MPCL). This model integrates a self-attention network, contrastive learning, and multitask training. The primary contributions of this study are outlined as follows.

1. To tackle the challenge of sparse data in short sequences, this paper integrates an informative augmentation module into the current self-supervised contrastive learning framework. This module addresses the above issue by reversing the order of the items within the short sequences and feeding them into a self-attention network for reverse pretraining. As a result of this process, virtual interaction items are generated for users with short sequences. These items are subsequently combined with the initial short sequences to form informative augmented sequences. This particular module enhances the information content of the short sequences while maintaining the original item correlations.
2. To address the generation of false-positive and false-negative samples in current contrastive learning frameworks, this paper employs data augmentation methods to produce multiple positive samples for contrastive learning purposes. Subsequently, a multipair contrastive loss is formulated using the generated positive and negative samples. It is important to highlight that for each sequence, n data augmentation instances are executed, yielding n positive samples. During alignment, the growth in the number of positive samples leads to a corresponding increase in the number of negative samples. This formulation of the multipair contrastive loss adeptly alleviates the adverse influence of false-positive and false-negative samples during network training.
3. Introducing an adaptive loss weighting Mechanism: This paper presents an adaptive loss weighting method within the context of multitask training. This method dynamically modifies the weight of the contrastive loss

during the training process, thereby expediting the model convergence process and augmenting the effectiveness of training.

The rest of this paper is organized as follows. Section 2 describes a literature review on sequential recommendation, attention mechanisms, and self-supervised contrastive learning and provides a research review. A detailed explanation of the proposed methodology is discussed in Sect. 3, including representation learning with informative augmentation, the self-attention network, the multipair contrastive loss, sequential recommendation, and multitask training. Section 4 discusses the utilized datasets, metrics, and baselines. Section 5 discusses the system requirements, the experimental settings adopted for training, a comparison among the results of the proposed methodology with those of the baselines, sensitivity experiments on the hyperparameters, ablation experiments, and comparison experiments conducted on datasets with different sparsity levels. Finally, the conclusion is provided in Sect. 6.

2 Related work

2.1 Sequential recommendation

Initially, sequential recommendation models primarily relied on Markov chains for sequence modelling. He et al. [1] combined Markov chains with similarity models to capture pairwise user-item and item-item interactions, enabling sparse sequential recommendation. The FPMC [6] model integrates matrix factorization (MF) with Markov chains to capture temporal information and long-term user preferences from sparse data. With the advancement of deep learning, recurrent neural networks (RNNs), such as long short-term memory (LSTM) and gated recurrent units (GRUs), have been employed for modelling user behaviour sequences. For instance, Hidasi et al. [7] utilized a GRU for session-based recommendation, while Wu et al. [8] explored both long-term and short-term item correlations using LSTM. Convolutional neural networks (CNNs) have also demonstrated effectiveness in terms of modelling short-term user interests. The Caser model proposed by Tang et al. [9] embeds recent item sequences into a temporal and spatial image, leveraging convolutional filters to learn sequence patterns as local image features. Furthermore, graph convolutional neural networks (GCNs) [10] and reinforcement learning [11] methods have shown remarkable performance in the field of sequential recommendation.

In recent years, inspired by the successful application of transformers in natural language processing (NLP) and computer vision (CV), transformer-based self-attention sequential recommendation models have emerged. These models have demonstrated unique advantages in capturing item correlations and modelling high-quality user preferences. Among them, SASRec [12] utilizes stacked transformer modules as a user sequence encoder to learn user preference representation vectors and model complex item correlations within the given sequence, ultimately performing the next-item prediction task. In 2019, Alibaba proposed the BST [13] model, which employs transformers to capture the sequential signals behind user behaviour sequences and has been deployed online on Taobao. While SASRec utilizes unidirectional transformers for encoding, Sun et al. argued that this unidirectional structure limits the learning capacity of the sequence representation process. Inspired by the BERT [14] model, they proposed a bidirectional encoding representation method for sequential recommendation [15] (BERT4Rec), achieving significant improvements. The LSSA [16] model utilizes transformers to learn both long-term and short-term user preferences, while the SSE-PT [17] model introduces a personalized transformer for personalized recommendation. CL4SRec [4] and CoSeRec [18] incorporate contrastive learning modules into transformer-based models to learn high-quality user preference representations. Overall, these transformer-based self-attention sequential recommendation models excel in capturing the complex features and long-term preferences within user behaviour sequences by learning user preference representations. They demonstrate good recommendation performance and scalability.

2.2 Attention mechanisms

Attention mechanisms have garnered substantial attention in recent years, particularly in the domains of natural language processing and recommendation systems. Attention techniques empower models to assign higher weights to the most crucial segments of the input, thereby significantly enhancing the expressive capacity levels of these models. In 2018, Alibaba introduced a deep interest network (DIN) [19] model, which integrates a local activation unit. This module employs an attention mechanism to evaluate the significance between the candidate item and the historical behavioural items. It dynamically computes the user's interest representation, thereby enabling personalized recommendations. Furthermore, the fusion of attention mechanisms with CNNs and RNNs has overcome their respective limitations in terms of capturing user interests. For instance, Tan et al. [20] devised a Bi-GRU neural network incorporating an attention mechanism to model both the long-term historical preferences and short-term consumption motivations of users. The introduced attention mechanism effectively captures the users' shifting interest towards the target items.

The self-attention mechanism, as a distinct attention mechanism form, has also been widely applied across various natural language processing tasks. It has proven successful in endeavours such as machine translation [3], sentiment analysis [21], and question answering [22]. The self-attention network has also delivered notable outcomes in the domain of sequential recommendation. For instance, the SASRec [12] model encompasses the entirety of user sequences through a streamlined and parallel self-attention mechanism, facilitating adaptive next-item predictions. The GC-SAN [23] harnesses the complementarity between self-attention networks and graph neural networks to achieve enhanced recommendation performance. In 2021, Zhao et al. [24] introduced a novel variational self-attention network (VSAN) for sequential recommendation, employing a variational reasoning paradigm [25] that integrates variational reasoning into the self-attention network to manage the uncertainty of user preferences. In 2023, Hao et al. [26] addressed explicit and implicit feature-level sequences in sequential recommendation. They introduced a feature-level deep self-attention network based on contrastive learning (FDSA-CL), which utilizes separate self-attention blocks on item-level sequences and feature-level sequences to model item transition patterns and feature transition patterns, respectively. This approach leads to significant performance improvements. Overall, attention mechanisms play a vital role in the research efforts dedicated to constructing efficient recommendation systems.

2.3 Self-supervised contrastive learning

Deep learning models encounter limitations when confronted with unlabelled sparse data, which has spurred research in the realm of self-supervised contrastive learning. This approach aims to uncover supervisory signals from unlabelled data to guide model training and has found extensive applications in diverse fields, including natural language processing, computer vision, graph embedding, and recommendation systems.

The CLEAR method, as proposed by Wu et al. [27], is a sentence-level contrastive learning approach that employs multiple sentence-level augmentation strategies. Its objective is to cultivate noise-invariant sentence representations, thereby enhancing the performance achieved in downstream tasks. SimCLR [28] utilizes a straightforward framework for contrastive visual representation learning, where the pivotal role of random image augmentation in defining effective prediction tasks is explored. Jiao et al. [29] introduced the Subg-Con method, which is a self-supervised representation learning approach based on subgraph contrast. This approach capitalizes on the robust correlations between central nodes and their sampled subgraphs to capture structural information. The method employs a contrastive loss defined on the subgraphs extracted from the original graph to learn node representations.

In recommendation systems, the integration of contrastive learning with deep learning-based recommendation models has demonstrably bolstered recommendation performance. Zhou et al. [30] conceived a contrastive self-supervised learning paradigm that leverages the principle of maximizing mutual information to enhance sequential recommendation models. Concerning data augmentation, SGL [31] utilizes three approaches on a user-item graph to modify its structure. Subsequently, the derived subgraphs serve as positive samples for contrastive learning. CL4SRec [4] and CoSeRec [18] employ augmentation methods tailored for sequential recommendation, facilitating the construction of positive samples for contrastive learning. In 2022, Hao et al. [32] presented a learnable model augmentation-based self-supervised learning framework for sequential recommendation (LMA4Rec). This framework utilizes model augmentation as a supplementary data augmentation method to generate diverse views. Subsequently, self-supervised learning is enacted between these contrastive views to extract self-supervised signals from the original sequence.

In terms of multipair contrastive learning, Tang et al. [33] introduced a multisample contrastive loss (MSCL). This loss mechanism balances the significance levels of positive and negative samples and calculates a contrastive loss employing multiple positive samples (as opposed to the use of a single positive sample in the BPR loss). This adaptation enhances the constructed graph model's ability to manage data sparsity. In 2022, Du et al. [34] proposed a multipair contrastive learning method founded on bidirectional transformers. This approach generates multiple high-quality positive samples through random masking and dropout for the subsequent contrastive learning step. Different from contrastive learning methods that rely on single-pair instances, this method exhibits superior performance and adaptability.

2.4 Research review

While transformer-based self-attention sequential recommendation models possess distinctive advantages in terms of capturing intricate user characteristics and long-term preferences, they are not exempt from the effects of data sparsity on recommendation performance. Furthermore, the absence of an efficient representation learning mechanism for users with concise sequences and restricted information significantly curtails the advancement of transformer-based self-attention sequential recommendation models.

The amalgamation of the self-supervised contrastive learning method and transformer-based self-attention sequential recommendation models elevates sequential recommendation to a new echelon. For instance, the CL4SRec model achieves an enhanced ability to withstand data sparsity to a certain degree. Nevertheless, sequential recommendation models founded on self-supervised contrastive learning still grapple with the task of formulating an effective learning mechanism for short user representation sequences. This limitation often results in suboptimal recommendations.

Furthermore, the issue of false-positive and false-negative samples within self-supervised contrastive learning methods, obtained by utilizing data augmentation operations, also impacts the resulting model's recommendation efficacy. Additionally, the existing research lacks a proven methodology for addressing this problem. Hence, this paper introduces the proposed informative augmentation module to rectify the inadequacies of prior studies concerning short sequence modelling. It further devises a multipair contrastive loss, enabling the model to effectively counteract the influences of false-positive and false-negative samples during the training process.

3 Sequential recommendation based on multipair contrastive learning with informative augmentation

The goal of sequential recommendation is to capture users' preferences from their historical behaviour sequences. However, users with short sequences possess fewer interaction items, which fail to furnish the model with sufficient information to learn precise preference representations, consequently influencing the resulting recommendation performance. Recently, data augmentation methods such as masking, cropping, and reordering have partially mitigated the impact of data sparsity on recommendation performance. Nonetheless, when applied to short sequences, these methods can potentially disrupt the order properties and underlying correlations among the items in these sequences. Therefore, we propose the IA-MPCL model to enrich the information contents of short sequences and enhance the representation learning process for users with abbreviated sequences. The model framework is depicted in Fig. 1a

The proposed model comprises two components: a representation learning module with informative augmentation and a self-supervised contrastive learning module for recommendation. In the representation learning module with informative augmentation, a self-attention network is utilized to conduct reverse pretraining on short sequences, yielding virtual interaction items for users with short sequences. These virtual interaction items are then combined with the original short sequences, resulting in the creation of new sequences. This process expands the information contents of the short sequences while retaining the original item correlations. Subsequently, data augmentation is applied to the informative augmented sequences (long sequences do not necessitate informative augmentation; data augmentation suffices), generating multiple augmented views for the learning of user preference representations.

In the segment focused on self-supervised contrastive learning for recommendation, we train the self-attention network to acquire high-quality user representations through the formulation of a recommendation loss and a multipair contrastive loss. The multipair contrastive loss regards n representations of the same user as positive samples for contrastive learning, effectively attenuating the impact of false-positive and false-negative samples on the

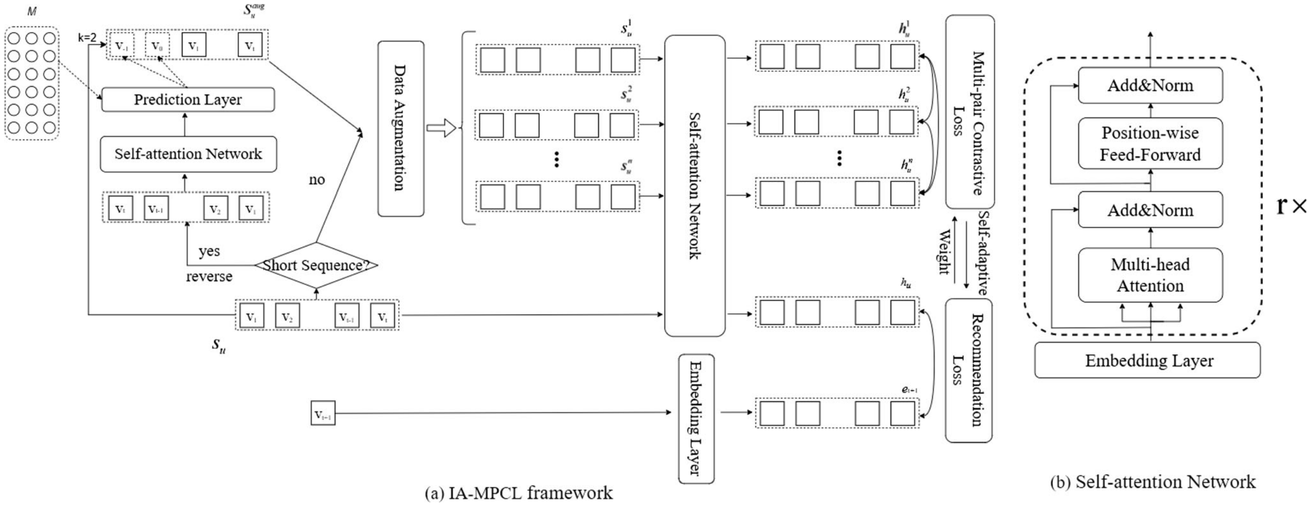


Fig. 1 Overall framework of the IA-MPCL model

self-attention network's training process. Ultimately, we introduce an adaptive loss weighting mechanism to dynamically adjust the contrastive loss weight within the comprehensive loss, enabling adaptable modifications during the training procedure of the self-attention network. The IA-MPCL model comprises four modules: (a) a representation learning module with informative augmentation, (b) a multipair contrastive loss module, (c) a sequential recommendation module, and (d) a multitask training module.

3.1 Representation learning with informative augmentation

The representation learning module with informative augmentation mainly comprises two mechanisms: informative augmentation for short sequences and representation learning for augmented views. This section is dedicated to short sequences within the given datasets, employing the reverse pretrained self-attention network to amplify the information contained in these short sequences. This process is followed by the generation of multiple augmented views for the short sequences via data augmentation methods. Ultimately, these views are fed into the self-attention network for representation learning.

Given that the encoder in our model is built upon a self-attention network, which is utilized in both the informative augmentation mechanism for short sequences and representation learning mechanism for augmented views, we initially introduce the embedding layer and self-attention network.

3.1.1 Embedding layer

The role of the embedding layer is to transform all input features into a fixed-length dense vector. Within the scope of this paper, the data fed into the embedding layer comprise the historical user behaviour sequences, which are denoted as $s_u = [v_1, v_2, \ldots, v_i, \ldots, v_m]$ with item ID features. Each item v_i in the sequence is mapped to a low-dimensional embedding vector $e_i \in \mathbb{R}^d$ using an item embedding matrix $\mathrm{E} \in \mathbb{R}^{T \times d}$. Here, T signifies the maximum length of the user sequence, and d represents the dimensionality of the embedding vector. In scenarios where the user sequence length surpasses T, we opt to truncate the most recent T interactions in chronological order to formulate their historical behaviour sequence. Conversely, for a sequence containing fewer than T interactions, we pad the sequence with "0" at the beginning until the sequence length reaches T.

To capture the positional order information of the items contained in the user behaviour sequences, a learnable position embedding matrix $\mathrm{P} \in \mathbb{R}^{T \times d}$ is utilized. Let $p_i \in \mathbb{R}^d$ be the position embedding vector for the i-th position in the sequence. Therefore, after the embedding layer, the user behaviour sequence $s_u = [v_1, v_2, \ldots, v_i, \ldots, v_m]$ is transformed into $\mathrm{E}(s_u) = [e_1 + p_1, e_2 + p_2, \ldots, e_i + p_i, \ldots, e_m + p_m]$, which serves as the input of the self-attention network.

3.1.2 Self-attention network

Upon traversing the embedding layer, the item sequences are subsequently input into the self-attention network for representation learning. The self-attention network, akin to the SASRec model, is fashioned by arranging multiple encoder modules consisting of transformers. Its architecture is outlined in Fig. 1b. A fundamental self-attention module encompasses multihead self-attention, normalization and residual connections, along with a position-wise feedforward network. Each component is elaborated upon in the ensuing descriptions.

3.1.2.1 Multihead self-attention To extract information from various subspaces at each position, a multihead self-attention mechanism is employed to handle the input embedding vectors. Initially, the input is subjected to h distinct linear transformations to map it to h different subspaces. Subsequently, the self-attention mechanism is implemented on each individual head. When given a hidden representation $H^l \in R^{(T \times d)}$ $H^l \in R^{(T \times d)}$ at the l-th layer, which functions as the input for the (l + 1)-th layer, the computational process of the multihead self-attention mechanism unfolds as follows:

$$\mathrm{Attention}(\mathrm{Q}, \mathrm{K}, \mathrm{V}) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d/h}}\right) \tag{1}$$

The equation above represents the scaled dot-product attention operation. Q, K, and V represent the query vector, the key matrix, and the value matrix, respectively. $\sqrt{d/h}$ is a scaling factor used to prevent the dot product from becoming too large.

$$\mathrm{head}_i = \mathrm{Attention}\left(H^l W_i^Q, H^l W_i^K, H^l W_i^V\right) \tag{2}$$

$$\mathrm{MultiHead}\left(H^l\right) = \mathrm{concat}(\mathrm{head}_1; \mathrm{head}_2; \ldots; \mathrm{head}_h) w^o \tag{3}$$

where $W_i^Q \in \mathbb{R}^{d \times d/h}$, $W_i^Q \in \mathbb{R}^{d \times d/h}$, $W_i^V \in \mathbb{R}^{d \times d/h}$, and $W^O \in \mathbb{R}^{d \times d}$ are the parameter matrices that need to be learned.

3.1.2.2 Normalization and residual connections Following the multihead self-attention calculation, normalization is implemented at the layer level, and a residual connection is introduced to mitigate potential degradation concerns within deep networks.

3.1.2.3 Feedforward network Taking the fact that the multihead self-attention mechanism primarily hinges on linear mappings into account, a feedforward network is incorporated to capture the nonlinear feature information within the input. This feedforward network comprises two fully connected layers accompanied by a ReLU activation function. The calculation expressions are provided below:

$$\mathrm{FFN}\left(h_i^l\right) = \mathrm{RELU}\left(h_i^l W_1 + b_1\right) W_2 + b_2 \quad (4)$$

$$\mathrm{FFN}\left(H^l\right)\mathrm{concat}(\mathrm{FFN}\left(h_1^l\right)^T;\ \ \mathrm{FFN}\left(h_2^l\right)^T; \ldots;\ \ \mathrm{FFN}\left(h_{|s_u|}^l\right)^T \quad (5)$$

where $W_1 \in \mathbb{R}^{d \times d}$, $W_2 \in \mathbb{R}^{d \times d}$, $b_1 \in \mathbb{R}^{d \times d}$, and $b_2 \in \mathbb{R}^{d \times d}$ are the learnable parameter matrices shared at each position.

Therefore, the relationship between the input $s_u = [v_1, v_2, \ldots, v_i, \ldots, v_m]$ and the output h_u of the user preference representation learning layer can be simplified as follows:

$$h_u = \mathrm{SAN}(s_u) \quad (6)$$

where $\mathrm{SAN}(\cdot)$ represents a self-attention network consisting of r stacked self-attention layers.

3.1.3 Informative augmentation of short sequences

The input short sequences contain limited interaction information, so even slight data augmentation perturbations can have significant impacts on the process of learning short sequence user representations. In the context of handling short sequences, a reverse data augmentation model called ASReP [35] has achieved successful practical experience, which inspires the research in this paper. We employ a reverse pretrained self-attention network to generate virtual interaction items for the short sequence users, artificially transforming the short sequences into "long sequences" to enhance their information contents. The specific generation process is described as follows:

First, the historical user behaviour sequence $s_u = [v_1, v_2, \ldots, v_m]$ is reversed to $\overleftarrow{s}_u = [v_m, v_{m-1}, \ldots, v_1]$. The reversed sequence $\overleftarrow{s}_u$ is then fed into the self-attention network for reverse learning, and the network's output yields the representation vector acquired from reverse learning. Subsequently, we apply a conventional latent factor model to the item set to compute the predicted score for item v_i, as depicted in the following formula:

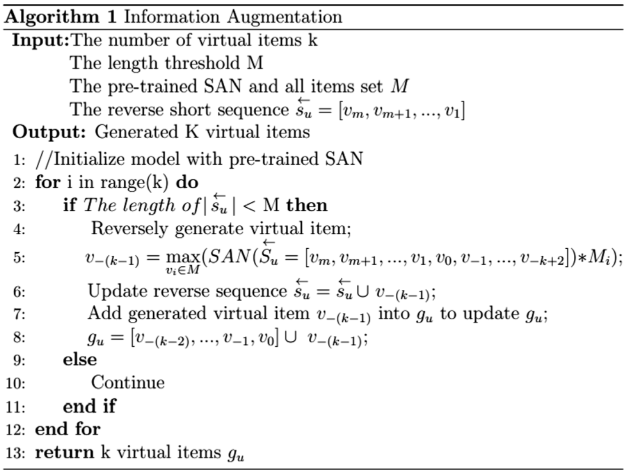
$$p_i = \underset{u}{\overleftarrow{H}} \times M_i \quad (7)$$

where p_i is the predicted score for item v_i, $\underset{u}{\overleftarrow{H}}$ represents the representation vector obtained from reverse learning for user u, and M_i is the latent vector for item v_i. The item with the highest predicted score is chosen as the subsequent item in the reverse prediction process, acting as the virtual interaction item for user u:

$$v_0 = \max_{v_i \in M} p_i \quad (8)$$

By concatenating the predicted v_0 at the beginning of the original sequence s_u, we obtain $\vec{s}_u = [v_0, v_1, \ldots, v_{m-1}, v_m]$. This reverse training procedure, which generates virtual interaction items, enhances the informational content of the short sequences while maintaining the semantic relevance of the original sequences.

The aforementioned process describes the generation of the first virtual item v_0 for a short sequence (the first iteration). Let k denote the number of generated virtual items, and let M denote the length threshold. By updating s_u with $\vec{s}_u$, we obtain $s_u = [v_0, v_1, v_2, \ldots, v_m]$. Reversing this sequence gives $\overleftarrow{s}_u = [v_m, v_{m-1}, \ldots, v_1, v_0]$. If $\left|\overleftarrow{s}_u\right| < \mathrm{M}$, we repeat the above iterative process to generate v_{-1} and update s_u and $\overleftarrow{s}_u$ using v_{-1}. This iterative procedure is repeated k times to obtain a set of virtual interaction items for user u: $g_u = [v_{-k+1}, \ldots, v_{-1}, v_0]$. Then, we insert g_u at the beginning of the original sequence s_u, resulting in the augmented sequence $s_u^{\mathrm{aug}} = [v_{-k+1}, \ldots, v_{-1}, v_0, v_1, v_2, \ldots, v_m]$ with $\left|s_u^{\mathrm{aug}}\right| < \mathrm{M}$. Setting a length threshold M for the informative augmented sequence prevents the number of virtual items from being too large, which would affect the accuracy of the representation learning process for short sequence users. The informative augmentation process for k virtual items using a pretrained self-attention network (SAN) is presented in Algorithm 1.



```
Algorithm 1 Information Augmentation
Input: The number of virtual items k
       The length threshold M
       The pre-trained SAN and all items set $M$
       The reverse short sequence $\overleftarrow{s}_u = [v_m, v_{m+1}, ..., v_1]$
Output: Generated K virtual items
//Initialize model with pre-trained SAN
for i in range(k) do
    if The length of $|\overleftarrow{s}_u| <$ M then
        Reversely generate virtual item;
        $v_{-(k-1)} = \max_{v_i \in M}(SAN(\overleftarrow{S}_u = [v_m, v_{m+1}, ..., v_1, v_0, v_{-1}, ..., v_{-k+2}]) * M_i)$;
        Update reverse sequence $\overleftarrow{s}_u = \overleftarrow{s}_u \cup v_{-(k-1)}$;
        Add generated virtual item $v_{-(k-1)}$ into $g_u$ to update $g_u$;
        $g_u = [v_{-(k-2)}, ..., v_{-1}, v_0] \cup v_{-(k-1)}$;
    else
        Continue
    end if
end for
return k virtual items $g_u$
```

3.1.4 Representation learning of augmented views

The primary objective of data augmentation is to furnish positive and negative samples for the creation of a contrastive loss. In this paper, we employ established data augmentation methods such as masking, cropping, and reordering to execute n rounds of data augmentation operations on the informative augmented sequence s_u^{aug} (for long sequences, data augmentation is applied directly). The formulation is stated as follows:

$$s_u^i = \underset{a_i \in \Phi}{\text{DA}}\left(s_u^{\text{aug}}\right) \tag{9}$$

where s_u^i is the i-th (i = 2, 3,…, n) augmented view; DA denotes the data augmentation operation; a_i is a specific data augmentation method; and Φ represents the set of methods, including masking, cropping, and reordering.

After obtaining n augmented views $\left[s_u^1, s_u^2, \ldots, s_u^n\right]$, we utilize a self-attention network to learn their corresponding representation vectors $\left[h_u^1, h_u^2, \ldots, h_u^n\right]$. The formulation is expressed as follows:

$$h_u^i = \text{SAN}\left(s_u^i\right) \tag{10}$$

where h_u^i represents the i-th representation vectors of the augmented views after completing representation learning.

3.2 Multipair contrastive loss

In CL4SRec, the authors considered two augmented views derived from the same user as positive pairs and two augmented views derived from different users as negative pairs. They employed a contrastive loss to enhance the consistency between the positive pairs and amplify the dissimilarity between the negative pairs. This contrastive loss guides the training process of the self-attention network. When constructing the contrastive loss, CL4SRec adopts a straightforward single-pair instance approach. Specifically, for a batch of N user behaviour sequences $\{s_u\}_{u=1}^N$, two data augmentation methods are randomly chosen for each s_u, resulting in 2N augmented views $\left\{s_1^i, s_1^j, s_2^i, s_2^j, \ldots, s_u^i, s_u^j, \ldots, s_N^i, s_N^j\right\}$. After performing representation learning, 2N representations are obtained: $\left\{h_1^i, h_1^j, h_2^i, h_2^j, \ldots, h_u^i, h_u^j, \ldots, h_N^i, h_N^j\right\}$. A representation pair $\left(h_u^i, h_u^j\right)$ from the same user is treated as a positive pair, while the remaining 2(N-1) samples are considered negative pairs with respect to h_u^i. The similarity between the representations is measured using the dot product, i.e., $\text{sim}(\text{A}, \text{B}) = A^T B$. Therefore, the contrastive loss L_{cl} used for training is defined as follows:

$$L_{cl}\left(h_u^i, h_u^j\right) = -\log \frac{e^{sim\left(h_u^i, h_u^j\right)}/\tau}{e^{sim\left(h_u^i, h_u^j\right)}/\tau + \sum_{k=1,k\neq u}^N \sum_{c\in(i,j)} e^{sim\left(h_u^i, h_k^j\right)}/\tau} \tag{11}$$

where τ is a temperature coefficient, which is a hyperparameter.

Existing studies have shown significant improvements in terms of guiding the model training process by using contrastive losses constructed with single positive pairs. However, the presence of false-positive and false-negative pairs affects model training, limiting the ability of contrastive learning to fully exploit the enormous potential of sequential recommendation models and consequently impacting the resulting recommendation accuracy.

Inspired by innovations in the multipair contrastive loss used for lightweight graph convolutional networks [33], this paper generates n augmented views (where $n > 2$) for each user behaviour sequence using data augmentation operations. This results in $n(n-1)/2$ positive pairs after completing representation learning. The purpose of this approach is to increase the number of positive pairs while also increasing the number of negative pairs. Subsequently, a multipair contrastive loss is constructed to mitigate the negative impacts of false-positive and false-negative pairs on model training. Specifically, for each user behaviour sequence s_u, after n rounds of data augmentation, n augmented views $\left[s_u^1, s_u^2, \ldots, s_u^n\right]$ are obtained. After applying the self-attention network for representation learning, each user produces n preference representations $\left[h_u^1, h_u^2, \ldots, h_u^n\right]$. The multipair contrastive loss L_{mpcl} can be defined as follows:

$$L_{\text{mpcl}} = \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| L_{cl}\left(h_u^i, h_u^j\right) \right\|_{i,j} \tag{12}$$

where $\left\| L_{cl}\left(h_u^i, h_u^j\right) \right\|_{i,j}$ is defined as a piecewise function, which is denoted as follows:

$$\left\| L_{cl}\left(h_u^i, h_u^j\right) \right\|_{i,j} = \begin{cases} L_{cl}\left(h_u^i, h_u^j\right) & i \neq j \\ 0 & i = j \end{cases} \tag{13}$$

It can be observed that apart from the difference in the numbers of samples used during training, the training process of multipair contrastive learning is identical to that of single-pair contrastive learning.

3.3 Sequential recommendation

The main task in this paper is sequential recommendation, which is performs next-item prediction based on a self-attention network. This task involves predicting the item that a user is most likely to interact with in the next time step when given their historical behaviour sequence

$s_u = [v_1, v_2, \ldots, v_t, \ldots, v_m]$. Mathematically, this task can be represented as follows:

$$\underset{v_i \in M}{\operatorname{argmax}} P(v_{m+1} = v_i | s_u) \tag{14}$$

where M represents the complete set of items that user u can interact with, and v_{m+1} represents the item that the user will interact with at the next time step.

To train the model, we utilize the negative log-likelihood loss to optimize the self-attention network. Let $s_u^t = [v_1, v_2, \ldots, v_t]$ be a subsequence of s_u that represents the user's historical behaviour sequence up to time step t. h_u^t denotes the user representation obtained by passing s_u^t through the self-attention network. The negative log-likelihood loss is used to optimize the self-attention network and can be defined as follows:

$$L_{\text{rec}}(h_u^t) = -\log(\sigma(h_u^t \cdot e_{t+1})) - \sum_{v_j \in [M - s_u]} \log(1 - \sigma(h_u^t \cdot v_j)) \tag{15}$$

where σ is the activation function, e_{t+1} represents the embedding vector of the item interacted with at time step t + 1,$[M - s_u]$ denotes the set of negative samples, and v_j represents a negatively sampled item from the negative sample set.

3.4 Multitask training

To acquire self-supervised signals from unlabelled raw data and integrate a contrastive learning task into the sequential recommendation process, this paper adopts a multitask training approach. Given that both tasks target the modelling of item correlations within user sequences, the utilization of multitask training can enhance the performance of the constructed sequential recommendation model. More specifically, this paper employs a linear combination of the recommendation loss and the contrastive loss to establish the multitask loss. Through this joint optimization process, the model is concurrently guided to learn both the sequential recommendation and contrastive learning tasks, leading to an overall performance improvement. The multitask loss is defined as follows:

$$L_{\text{multi}} = L_{\text{rec}} + \delta L_{\text{mpcl}} \tag{16}$$

where δ is a weight parameter that is used to control the importance of the contrastive loss during training.

In the existing methods, multitask training is commonly employed. However, the weights assigned to each task remain constant throughout the iterative convergence process, impeding model convergence and constraining the auxiliary impact of the contrastive loss. To expedite the convergence procedure and attain optimality, this paper introduces a recursive approach to adaptively update the weights. Specifically, the weight of the contrastive loss is dynamically adjusted during the iterative process. The specific details are as follows:

$$\delta_{d+1} = \omega \delta'_{d+1} + (1 - \omega)\delta_d \tag{17}$$

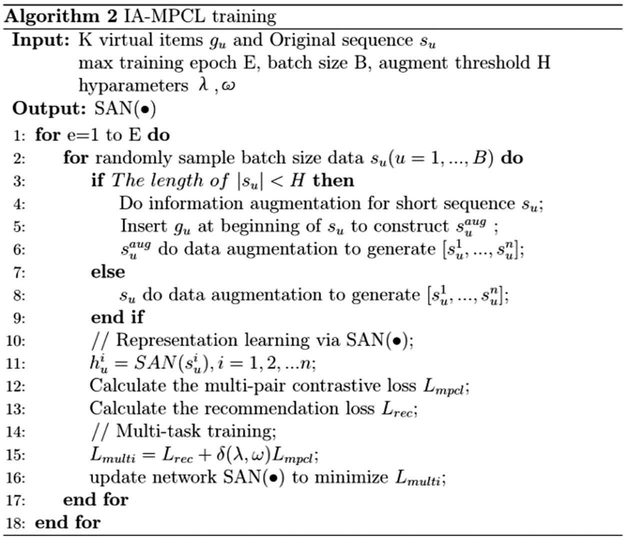
$$\delta'_{d+1} = \frac{L_{(d+1)\text{rec}}}{L_{(d+1)\text{rec}} + \lambda L_{(d+1)\text{mpcl}}} \tag{18}$$

In the above equations, d represents the d-th iteration, ω is the parameter for learning the δ value between two consecutive iterations, λ is the significance indicator for the contrastive loss, and $L_{(d+1)\text{rec}}$ and $L_{(d+1)\text{mpcl}}$ represent the recommendation loss and contrastive loss, respectively, after the $(d + 1)$-th iteration. The initial value of δ is set to 0 and is updated at the end of each iteration during the training process. Therefore, the multitask loss with dynamically updated weights can be updated as follows:

$$L_{(d+1)\text{multi}} = L_{(d+1)\text{rec}} + \delta_{d+1} L_{(d+1)\text{mpcl}} \tag{19}$$

Algorithm 2 summarizes the training process of IA-MPCL,

which optimizes the parameters of the self-attention network via multitask training.



Algorithm 2 IA-MPCL training

```
Input: K virtual items $g_u$ and Original sequence $s_u$
       max training epoch E, batch size B, augment threshold H
       hyparameters $\lambda$ ,$\omega$
Output: SAN(•)
for e=1 to E do
    for randomly sample batch size data $s_u(u = 1, ..., B)$ do
        if $The\ length\ of\ |s_u| < H$ then
            Do information augmentation for short sequence $s_u$;
            Insert $g_u$ at beginning of $s_u$ to construct $s_u^{aug}$ ;
            $s_u^{aug}$ do data augmentation to generate $[s_u^1, ..., s_u^n]$;
        else
            $s_u$ do data augmentation to generate $[s_u^1, ..., s_u^n]$;
        end if
        // Representation learning via SAN(•);
        $h_u^i = SAN(s_u^i), i = 1, 2, ...n$;
        Calculate the multi-pair contrastive loss $L_{mpcl}$;
        Calculate the recommendation loss $L_{rec}$;
        // Multi-task training;
        $L_{multi} = L_{rec} + \delta(\lambda, \omega) L_{mpcl}$;
        update network SAN(•) to minimize $L_{multi}$;
    end for
end for
```

4 Experimental settings

4.1 Datasets

This study evaluates the performance of the proposed model using two subsets of the Amazon review dataset: Beauty (5-core) and Sports_and_Outdoors (5-core). The detailed statistical information of the datasets is presented in Table 1.

Table 1 Statistics of the datasets

Datasets	Users	Items	Actions	Avg. length	Proportion (%)
Sports	35,598	18,357	296,337	8.3	82.2
Beauty	22,363	12,101	198,502	8.8	81.0

We treat each review in both datasets as a user-item interaction and then, arrange the item IDs of the same user in chronological order. This allows us to obtain an interaction sequence for each user. During data processing, we define users with sequence lengths below 10 as short sequence users. The proportions in Table 1 represent the percentages of users with sequence lengths less than 10 among the total numbers of users. The statistics show that over 80% of the sequences have lengths that are less than 10. Figure 2 illustrates the distribution of the user counts for different sequence length intervals, indicating that short sequences dominate in both datasets.

4.2 Metrics

This study employs rank-based evaluation metrics to assess the effectiveness of the model, including the hit ratio (HR@K) and the normalized discounted cumulative gain (NDCG@K), where K is set to 5, 10, and 20.

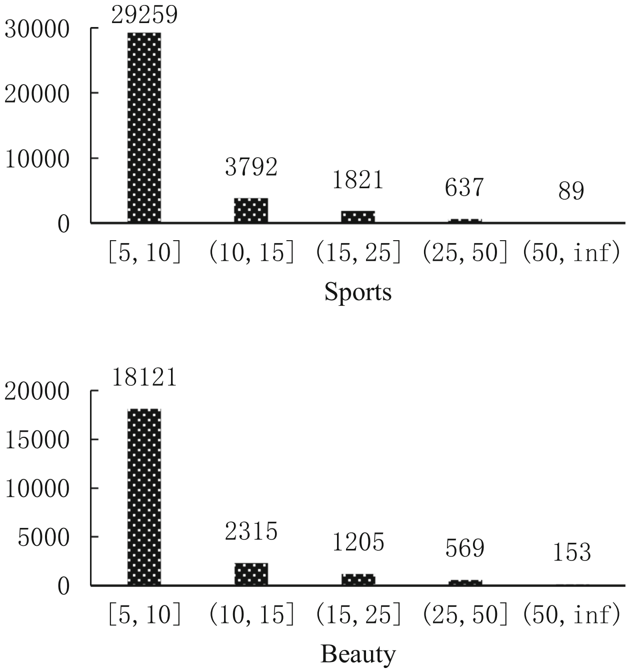


Fig. 2 Sequence length distributions of the two utilized datasets

The hit ratio (HR) is a commonly used metric for measuring the recall of top K recommendation results, as shown in Formula (20):

$$\mathrm{HR@K} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{hits}(i) \tag{20}$$

where N represents the total number of users and hits(i) indicates whether the value accessed by the i-th user is among the top-K recommended items (1 if true and 0 otherwise).

The normalized discounted cumulative gain (NDCG) is used to measure the precision of the item rankings in the top K recommendations, as shown in Formula (21):

$$\mathrm{NDCG@K} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\log_2(p_i+1)} \tag{21}$$

where N represents the total number of users and p_i denotes the position of the true accessed value of the i-th user in the top K recommendation list. If the recommended list does not contain this value, $p_i \to \infty$.

4.3 Baselines

The model proposed in this paper belongs to the category of sequential recommendation models. To validate its effectiveness, we compare it with three other sequential recommendation models.

1. *Caser* This model utilizes a convolutional neural network (CNN) to extract information from short-term sequences, emphasizing the impact of short-term information on sequence modelling.
2. *SASRec* This is a sequential recommendation model that models user sequences by stacking self-attention modules to capture dynamic user interests. SASRec has served as the foundational model for many subsequent research studies on sequential recommendation. The user sequence encoding part of our model is also inspired by SASRec.
3. *CL4SRec* This approach incorporates self-supervised contrastive learning into transformer-based sequential recommendation models. It utilizes random augmentation techniques to generate positive views, enhancing the robustness of sequential recommendation models against data sparsity.

5 Experiments and results analysis

The experimental setup for this study involves a Windows operating system, an i5 processor, and Python 3.7. All experiments are implemented using PyTorch, with the

Adam optimizer used for weight and parameter updates. The batch size is set to 256, and the learning rate is set to 0.001. The embedding dimensionality is set to 64, and the maximum sequence length is set to 50. In the self-attention network, the number of heads for the multihead attention mechanism is set to 2.

Negative sampling, as a crucial aspect of recommendation model training, significantly affects the training effectiveness of the developed model. Commonly used negative sampling methods include random negative sampling, popularity-based negative sampling, and model-based negative sampling. In the experiments conducted in this paper, negative samples are acquired for the recommendation loss through randomization. This entails treating each item within the complete set of items with equal importance and randomly sampling them with an equal probability. The utilization of random negative sampling in this paper is aimed at enhancing the efficiency of training and preventing the introduction of new biases during the sampling process. Moreover, existing self-supervised contrastive learning frameworks, such as CL4SRec, also employ random negative sampling. This is done to facilitate a more straightforward comparison and to underscore the advantages of our proposed innovations. Thus, we adhere to the negative sampling setup used in these existing methods.

When constructing the multipair contrastive loss, negative samples are derived from the representations of different users. Specifically, two samples acquired from distinct users are considered a negative pair.

5.1 Overall performance comparison

To validate the effectiveness of the proposed model, we conduct experiments on two publicly available datasets, comparing IA-MPCL with the three aforementioned sequential recommendation models. The experimental results are presented in Table 2.

Based on the experimental results, we can observe that the following.

First, the Caser model performs poorly on both datasets. Compared to transformer-based models, its sequence learning approach based on a CNN seems to be less effective. This suggests that transformer-based models have a greater ability to capture the preference information hidden in user behaviour sequences.

Second, CL4SRec, which incorporates self-supervised contrastive learning into the SASRec model to address the data sparsity issue, demonstrates significant advantages in terms of mitigating the negative impact of data sparsity on user preference representations. The recommendation accuracy (NDCG@10) of this model is improved by 13.8% and 11.0% on the Beauty and Sports datasets, respectively, indicating the benefits of contrastive learning tasks.

Finally, IA-MPCL outperforms the two transformer-based sequential recommendation models on both datasets. Compared to SASRec, IA-MPCL achieves average improvements of 21.6% and 27.4% on the two datasets, respectively. SASRec fails to effectively address the impact of data sparsity on representation modelling, resulting in inferior performance compared to that of IA-MPCL. Compared to CL4SRec, IA-MPCL achieves average improvements of 8.5% on the Beauty dataset and 14.3% on the Sports dataset. IA-MPCL provides feasible solutions for addressing the cold-start problem and the false-positive and false-negative sample issues in CL4SRec, and the comparative results indicate that IA-MPCL effectively alleviates the negative impacts of these issues on the accuracy of representation learning.

5.2 Performance comparison conducted on datasets with different sparsity levels

Short sequences are indicative of data sparsity. To validate the effectiveness of our proposed model on extremely sparse datasets, we conduct experiments on datasets with varying sparsity levels. We randomly select a certain

Table 2 Comparison among different models

Datasets	Baseline	HR@5	NDCG@5	HR@10	NDCG@10	HR@20	NDCG@20
Beauty	Caser	0.0251	0.0145	0.0342	0.0226	0.0643	0.0298
	SASRec	0.0356	0.0231	0.0582	0.0303	0.0902	0.0384
	CL4SRec	0.0401	0.0268	0.0642	0.0345	0.0974	0.0428
	IA-MPCL	0.0461	0.0295	0.0694	0.0373	0.1028	0.0457
	Improv	15.0%	10.1%	8.1%	8.1%	5.5%	6.8%
Sports	Caser	0.0154	0.0114	0.0261	0.0135	0.0399	0.0178
	SASRec	0.0206	0.0135	0.0320	0.0172	0.0497	0.0216
	CL4SRec	0.0231	0.0146	0.0369	0.0191	0.0557	0.0238
	IA-MPCL	0.0270	0.0175	0.0411	0.0221	0.0606	0.0270
	Improv	16.9%	19.9%	11.4%	15.7%	8.8%	13.4%

percentage of short sequence users from the Sports dataset and categorize the dataset into different sparsity levels: dense data (with short sequence users accounting for 60.6% of the total), moderately dense data (with short sequence users accounting for 67.5% of the total), moderately sparse data (with short sequence users accounting for 75.5% of the total), and extremely sparse data (the original dataset, with short sequence users accounting for 82.2% of the total). The statistical information of these datasets is presented in Table 3.

With experimental settings involving 5 virtual items, 2 self-attention network layers, and 4 positive samples for contrastive learning, we compare the performances achieved by IA-MPCL on datasets with varying sparsity levels. The experimental results are presented in Table 4. It is evident that as the proportion of short sequence users increases and the dataset becomes sparser while growing larger in size, IA-MPCL demonstrates enhanced recommendation performance. Notably, IA-MPCL achieves the best results on extremely sparse datasets. This observation underscores the robustness of the proposed model with respect to combating data sparsity.

5.3 Parameter sensitivity

We examine the influences of various hyperparameters on IA-MPCL's performance across the two datasets. Specifically, we conduct experiments on three pivotal hyperparameters: the number of virtual items (k), the number of positive samples for contrastive learning (n) and the number of layers in the self-attention network (r). To maintain control over the variables, we alter only one hyperparameter at a time while retaining the other hyperparameters at their optimal values.

5.3.1 The number of virtual items

Virtual items are latent interaction items generated for short sequence users when using the self-attention network, and they serve as informative augmentation items. The number of virtual items (k) directly affects the accuracy of short sequence user representation learning. Too few virtual items do not provide sufficient informative augmentation, while too many virtual items directly affect the correlations among the original items in the sequence, resulting in negative effects. In this experiment, we select different numbers of virtual items to observe the resulting model performance differences. For the Beauty dataset, we set the number of positive samples for contrastive learning (n) to 4 and the number of layers in the self-attention network (r) to 1. For the Sports dataset, we set n to 4 and r to 2. The other parameters are kept at their optimal values for both datasets. The experimental results, as shown in Fig. 3, indicate that IA-MPCL achieves the best performance when k is set to 5 for both datasets. When k exceeds 5, the recommendation performance shows varying degrees of decline.

5.3.2 The number of positive samples

The number of positive samples (n) is used to adjust the number of positive pairs in the contrastive loss. Different numbers of positive and negative pairs have varying effects

Table 4 Performance comparison results obtained on datasets with different sparsity levels

Datasets	HR20	NDCG20
Dense data	0.0528	0.0227
Moderately dense data	0.0559	0.0250
Moderately sparse data	0.0578	0.0255
Extremely sparse data	0.0606	0.0270

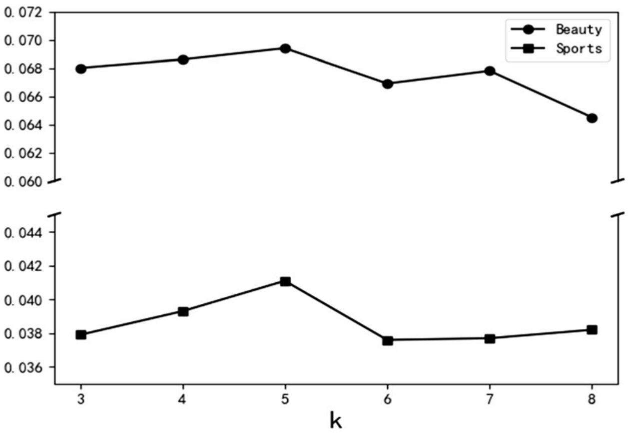


Fig. 3 Recommendation performance achieved with different k values on two datasets

Table 3 Statistics of the datasets with varying sparsity levels

Datasets	Users	Items	Actions	Avg.length	Proportion (%)
Dense data	14,482	18,162	154,720	10.7	60.6
Moderately Dense data	19,505	18,324	193,503	9.9	67.5
Moderately sparse data	23,260	18,351	210,547	9.0	75.5
Extremely sparse data	35,598	18,357	296,337	8.3	82.2

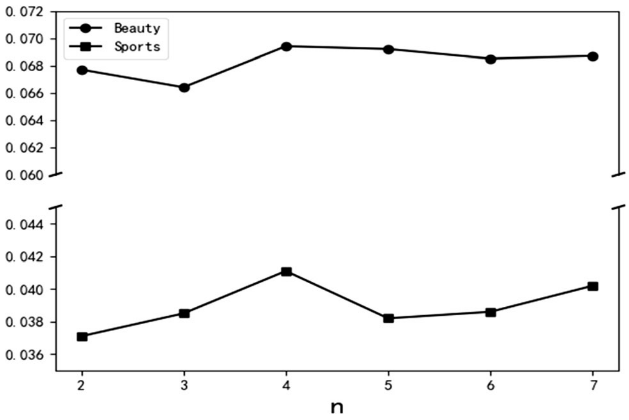


Fig. 4 Recommendation performance achieved with different n values on two datasets

on mitigating the impacts of false-positive and false-negative samples during training. Figure 4 shows the recommendation performance achieved by IA-MPCL under different values of *n*. The optimal recommendation performance is achieved when *n* is set to 4. Increasing the number of positive samples can significantly improve the recommendation performance. However, having more positive samples is not always better, as there is an upper limit. From the Beauty dataset, it can be seen that after the number of positive samples exceeds 4, further increasing the number does not lead to performance improvements, and the performance reaches a stable state.

5.3.3 The number of layers

The number of layers (*r*) in the self-attention network also influences the model's capacity to capture user preference information across sequences. Thus, we investigate the impacts of different numbers of layers in the self-attention network on sequence modelling, aiming to identify the optimal number of layer for the model. Figure 5 presents the experimental outcomes obtained on both datasets. Notably, a higher number of layers does not necessarily yield better performance. Although additional layers significantly extend the training time, they do not enhance accuracy. On the Beauty dataset, the model excels when r is set to 1, while other layer counts result in varying degrees of performance decline. On the Sports dataset, the optimal recommendation performance is achieved when r is set to 2. The results obtained on both datasets imply that a lower number of layers in the self-attention network structure aids in learning more intricate item correlations and capturing more precise user preferences.

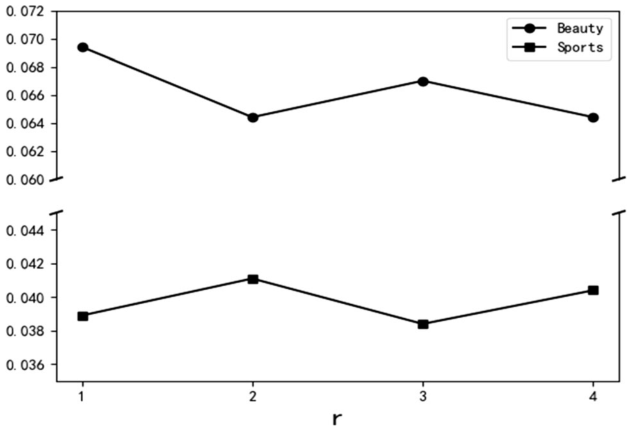


Fig. 5 Recommendation performance achieved with different r values on two datasets

5.4 Ablation study

We conduct ablation experiments to assess the necessity of the informative augmentation module and the multipair contrastive loss in the IA-MPCL model. By excluding the informative augmentation module, not employing the multipair contrastive loss, and not utilizing the adaptive loss weighting mechanism, the model regresses to CL4SRec, which acts as the baseline model. Subsequently, we integrate the informative augmentation module and the multipair contrastive loss into the baseline model. Through a performance comparison among these four models, the experimental results presented in Table 5 are obtained.

Based on the experimental results, the models with the informative augmentation module achieve improved recommendation performance on both datasets in comparison with the baseline model. This clearly demonstrates the effectiveness of this module in alleviating the impact of the data sparsity issue in short sequences on user representation modelling. Moreover, the models using the multipair contrastive loss also exhibit improvements over the baseline model. The baseline model utilizes a contrastive loss with single positive pairs, which fail to effectively address the issues of false-positive and false-negative samples during training. By incorporating the multipair contrastive loss, the model training process is optimized, resulting in a

Table 5 Results of ablation experiments

	Beauty	Sports
Basic model	0.0401	0.0231
+Informative augmentation	0.0422	0.0247
+Multipair contrastive learning	0.0452	0.0264
IA-MPCL	0.0461	0.0270

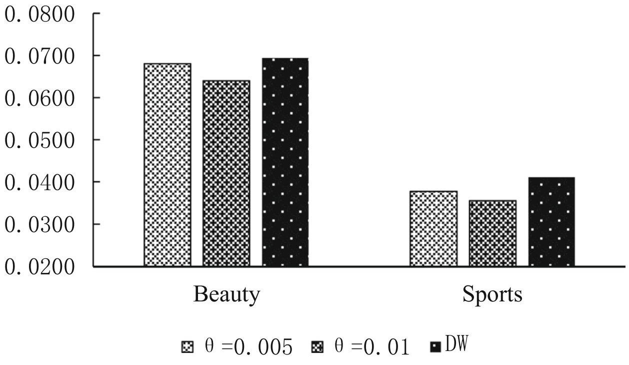


Fig. 6 Results of the ablation experiment concerning the adaptive loss weight

12.7% recommendation accuracy increase. This indicates that the multipair contrastive loss can effectively mitigate the negative impacts of false-positive and false-negative samples on model training and yield enhanced recommendation performance.

To evaluate the impact of the dynamic loss weighting mechanism on model training, this study replaces the dynamic loss weighting (DW) operation in IA-MPCL with fixed weights. Two fixed weight values, 0.005 and 0.01, are chosen for comparison with IA-MPCL. The experimental results are presented in Fig. 6.

During the experiments, we observe that the contrastive loss is often significant in the early stages of training and can decrease in the later stages. If a fixed weight value is used, the model would require more time to converge due to the utilization of a larger contrastive loss in the early stages, and the model might not effectively prioritize the contrastive learning task in the later stages. Therefore, this study employs a dynamic loss weighting mechanism that adjusts the value of the contrastive loss weight based on the magnitude of the contrastive loss during training. From the results of the ablation experiments, it is evident that the model with the dynamic loss weighting mechanism achieves the best recommendation performance on both datasets. Setting the contrastive weight loss to either 0.01 or 0.005 does not yield comparable results.

6 Conclusion

The existing sequential recommendation models face challenges related to data sparsity and noise, including the negative effects of false-positive and false-negative samples during contrastive learning, as well as the fixed weight of the contrastive loss, which hampers the convergence speed and training effectiveness of the constructed model. Addressing these issues, this paper presents a sequential recommendation model based on multipair contrastive learning with informative augmentation. This model combines a self-attention network with contrastive learning within a multitask learning framework.

First, to tackle the data sparsity issue in short sequences, this study employs reverse sequences to train the self-attention network, predicting potential user interaction items and incorporating them at the beginning of the original sequences. This augmentation enhances the information contents of short sequence users, improving the accuracy of their interest representations. Second, to overcome the problems associated with false-positive and false-negative samples during the training process of contrastive learning, this paper generates multiple positive samples through data augmentation operations. A multipair contrastive loss is formulated along with corresponding increased negative samples, effectively mitigating the impacts of false-positive and false-negative samples on the effectiveness of training. Finally, an adaptive loss weighting mechanism dynamically adjusts the contrastive loss weight based on the magnitudes of the recommendation loss and the contrastive loss, optimizing the self-attention network training process. The paper conducts a comprehensive series of experiments on two public datasets. These experiments include comparisons with baseline models, evaluations on datasets with varying sparsity levels, ablation experiments, and hyperparameter analyses. The results affirm the effectiveness of the proposed recommendation model, particularly its robustness against data sparsity.

This study addresses the challenges of the existing self-supervised contrastive learning frameworks, improving contrastive learning algorithms for recommendation systems. Future research may explore the deeper integration of contrastive learning methods with the self-attention network to enhance the recommendation model's anti-interference capabilities, thus boosting the accuracy and robustness of recommendation systems.

Funding This work was supported by Shanghai Philosophy and Social Science Planning Project (2022ZGL010). The financial support is gratefully acknowledged.

Data availability statement Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. He R, McAuley J (2016) Fusing similarity models with markov chains for sparse sequential recommendation. In: 2016 IEEE 16th

international conference on data mining (ICDM). IEEE, pp 191–200

2. Balázs H, Massimo Q, Alexandos K et al (2016) Parallel recurrent neural network architectures for feature-rich session-based recommendations. In: Proceedings of the 10th ACM conference on recommender systems, pp 241–248
3. Ashish V, Noam S, Niki P et al (2017) Attention is all you need. Adv Neural Inf Process Syst 30
4. Xie X, Sun F, Liu Z et al (2020) Contrastive learning for sequential recommendation. arXiv preprint arXiv:2010.14395
5. Qiu R, Huang Z, Yin H et al (2022) Contrastive learning for representation degeneration problem in sequential recommendation. In: WSDM., pp 813–823
6. Rendle S, Freudenthaler C, Schmidt-Thieme L (2010) Factorizing personalized markov chains for next-basket recommendation. In WWW, pp 811–820
7. Hidasi B, Karatzoglou A, Baltrunas L et al (2016) Session-based recommendations with recurrent neural networks. arXiv preprint arXiv:1511.06939
8. Wu C-Y, Ahmed A, Beutel A et al (2017) Recurrent recommender networks. In: WSDM, pp 495–503
9. Tang J, Wang K (2018) Personalized top-N sequential recommendation via convolutional sequence embedding. In: WSDM, pp 565–573
10. He X, Deng K, Wang X et al (2020) Lightgcn: Simplifying and powering graph convolution network for recommendation. In: SIGIR, ACM, pp 639–648
11. Guo P, Xiao K, Ye Z et al (2022) Intelligent career planning via stochastic subsampling reinforcement learning. Sci Rep 12(1):8332
12. Kang W-C, McAuley J (2018) Self-attentive sequential recommendation. In ICDM. IEEE, pp 197–206
13. Chen Q, Zhao H, Li W et al (2019) Behavior sequence transformer for e-commerce recommendation in Alibaba. In: Proceedings of the 1st International workshop on deep learning practice for high-dimensional sparse data, pp 1–4
14. Devlin J, Chang M-W, Lee K et al (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
15. 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, pp 1441–1450
16. Chengfeng Xu, Feng J, Zhao P et al (2021) Long- and short-term self-attention network for sequential recommendation. Neurocomputing 423(2021):580–589
17. Wu L, Li S, Hsieh C-J et al (2020) SSE-PT: sequential recommendation via personalized transformer. In: RecSys. ACM, pp 328–337
18. Liu Z, Chen Y, Li J et al (2021) Contrastive Self-supervised sequential recommendation with robust augmentation. arXiv preprint arXiv:2108.06479
19. Zhou G, Zhu X, Song C et al (2018) Deep interest network for click-through rate prediction. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery and data mining, pp 1059–1068
20. Tan Q, Liu F (2019) Recommendation based on users' long-term and short-term interests with attention. Math Prob Eng 2019:1–13
21. Lin Z, Feng M, dos Santos CN, et al (2017) A structured self-attentive sentence embedding. arXiv preprint arXiv:1703.03130
22. Li X, Song J, Gao L et al (2019) Beyond rnns: positional self-attention with co-attention for video question answering. In: Proceedings of the AAAI conference on artificial intelligence, vol 33, issue 01, pp 8658–8665
23. Xu C, Zhao P, Liu Y et al (2019) Graph contextualized self-attention network for session-based recommendation. IJCAI 19:3940–3946
24. Zhao J, Zhao P, Zhao L et al (2021) Variational self-attention network for sequential recommendation. In: 2021 IEEE 37th international conference on data engineering (ICDE). IEEE, pp 1559–1570
25. Kingma DP, Rezende DJ, Mohamed S et al (2014) Semi-supervised learning with deep generative models. Adv Neural Inf Process Syst 27
26. Hao Y, Zhang T, Zhao P et al (2023) Feature-level deeper self-attention network with contrastive learning for sequential recommendation. IEEE transactions on knowledge and data engineering
27. Wu Z, Wang S, Gu J et al (2020) CLEAR: contrastive learning for sentence representation. arXiv preprint arXiv:2012.15466
28. Chen T, Kornblith S, Norouzi M et al (2020) A simple framework for contrastive learning of visual representations. In: International conference on machine learning. PMLR, pp 1597–1607
29. Jiao Y, Xiong Y, Zhang J et al (2020) Sub-graph contrast for scalable self-supervised graph representation learning. In: 2020 IEEE international conference on data mining (ICDM). IEEE, pp 222–231
30. Zhou K, Wang H, Zhao WX 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 and knowledge management, pp 893–1902
31. Wu J, Wang X, Feng F et al (2021) Self-supervised graph learning for recommendation. In: Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval, pp 726–735
32. Hao Y, Zhao P, Xian X et al (2022) Learnable model augmentation self-supervised learning for sequential recommendation. arXiv preprint arXiv:2204.10128
33. Tang H, Zhao G, Wu Y et al (2021) Multisample-based contrastive loss for top-k recommendation. IEEE Transactions on Multimedia
34. Du H, Shi H, Zhao P et al (2022) Contrastive learning with bidirectional transformers for sequential recommendation. In: Proceedings of the 31st ACM international conference on information and knowledge management, pp 396–405
35. Liu Z, Fan Z, Wang Yet al (2021) Augmenting sequential recommendation with pseudo-prior items via reversely pre-training transformer. In: Proceedings of the 44th international ACM SIGIR conference on Research and development in information retrieval, pp 1608–1612

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.