Personalized Representation With Contrastive Loss for Recommendation Systems

Hao Tang, Guoshuai Zhao, *Member, IEEE*, Jing Gao, and Xueming Qian, *Member, IEEE*

***Abstract*—Sequential recommendation mines the user's interaction sequence or time information to get better recommendations and thus is gaining more and more attention. Existing sequential recommendations tend to build new models, and the study of the loss function is seriously neglected. Despite the increasing attention paid to contrastive learning recently, we believe that the key to contrastive learning is contrastive loss(CL), which also provides a new option for sequential recommendation. However, we find it works against the personalized representation of features. First, it is a relative constraint that keeps positive and negative samples away from each other but without an absolute constraint. Second, recent studies have shown that all embeddings should be uniformly distributed. However, CL only widens the distance of positive and negative samples within the training batch, rather than making a uniform distribution of all items. These two shortcomings make the embedding space too compact, which is harmful to personalized representation and recommendation. Therefore, this article proposes Personalized Contrastive Loss (PCL) to combine CL with absolute constraints of BCE/CE and employs regularization methods to make the representations uniformly distributed. State-of-the-art results are obtained in experiments on several commonly used datasets. The code and data will be available on GitHub.**

***Index Terms*—Personalization, contrastive loss, sequential recommendation, uniformity.**

I. Introduction

IN THE era of Big Data, recommendation systems are widely studied to face information overload and information explosion problems [1], [2], [3]. Sequential recommendation considering temporal information and order of user interaction is one of the important directions [4], [5], [6], [7]. It is already widely used in various recommendations, such as music recommendations [8], fashion recommendations [9], personalized recommendations [10], etc. Different methods have been proposed for sequential recommendation, which can be divided into two categories. One category is the traditional sequential recommendation algorithms, which generally use classical algorithms such as collaborative filtering and Markov chains. The other category is the deep neural network-based sequential recommendation methods [4], [11], [12], [13]. Especially, self-attention mechanisms or Transformer Encoder have been introduced into this field and significantly facilitated the sequential recommendation [5], [6], [14], [15]. Recently, the contrastive learning-based sequence recommendation method has also become one of the hot directions of research [7], [16], [17], [18]. However, these studies are limited to contrastive learning frameworks.

We argue that the key of contrastive learning is contrastive loss(CL) which also has the potential for sequential recommendations. Various studies have focused on how to model sequential recommendations. The available loss functions for sequential recommendation are very few and are rarely studied. The commonly used loss functions are almost always Cross Entropy(CE) and Binary Cross Entropy (BCE), and Bayesian Personalized Ranking (BPR) is sometimes used. And the CL-related MSCL [19], RCL [20] and SSM [21] have achieved better results for top-k recommendations. Therefore, it is meaningful to study the contrastive loss-based function for sequential recommendations.

Personalization is particularly important for sequential recommendations. Because sequential recommendations specifically require that the users' representations at different positions or times in the sequence are different and personalized. So personalized feature representation is the key to achieving accurate recommendations. And the user's representation at different time is obtained by feature aggregation of items' embeddings in more and more methods so personalization is also important for items' representation. Following DuoRec [16], when embeddings are projected into 2D by SVD with colors indicating the frequency of items, we found that the feature space under CL is too compact as shown in the left of Fig. 1, which is clearly detrimental to the individualized representation of features. The variance of embeddings can be used to measure the personalization of features [22]. Our experiments show that the embeddings' variance of CL is only 1/3 of the variance of BCE loss on ML-1 M dataset.

Manuscript received 11 October 2022; revised 9 March 2023 and 16 May 2023; accepted 29 June 2023. Date of publication 14 July 2023; date of current version 2 February 2024. This work was supported in part by NSFC, China under Grant 61902309, in part by ShaanXi Province under Grant 2018JM6092, in part by the Fundamental Research Funds for the Central Universities, China under Grants xxj022019003 and xzd012022006, in part by China Postdoctoral Science Foundation under Grant 2020M683496, in part by the National Postdoctoral Innovative Talents Support Program, China under Grant BX20190273, and in part by the Science and Technology Program of Xi'an, China under Grant 21RGZN0017. The Associate Editor coordinating the review of this manuscript and approving it for publication was Prof. Liqiang Nie. *(Corresponding author: Guoshuai Zhao.)*

Hao Tang is with the School of Information and Communication Engineering, Xi'an Jiaotong University, Xi'an 710049, China, and also with China Unicom Shaanxi Branch, Xi'an 710075, China (e-mail: th1002@stu.xjtu.edu.cn).

Guoshuai Zhao and Jing Gao are with the School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China, and also with Shaanxi Yulan Jiuzhou Intelligent Optoelectronic Technology Company, Ltd, Xi'an 710049, China (e-mail: guoshuai.zhao@xjtu.edu.cn; gaojing0423@stu.xjtu.edu.cn).

Xueming Qian is with the Ministry of Education Key Laboratory for Intelligent Networks and Network Security, School of Information and Communication Engineering, and SMILES LAB, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: qianxm@mail.xjtu.edu.cn).

Digital Object Identifier 10.1109/TMM.2023.3295740

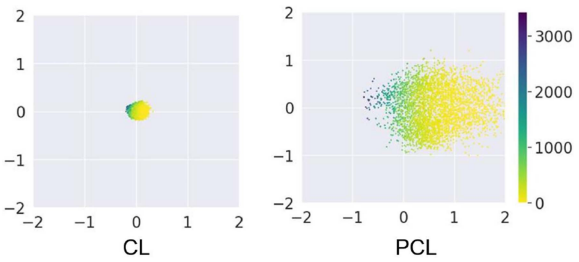


Fig. 1. Item embeddings of ML-1 M dataset in the feature space, with CL on the left and our proposed PCL on the right. We significantly expand the distance between items and the whole feature space, which facilitates the personalized representation of features.

Therefore, CL should be improved for the personalized feature representation problem. There are two reasons for CL makes a compact embedding space in sequential recommendations:

1) In CL, the closer the user is to the positive sample, the better, and the further the user is from the negative sample, the better. This is a relative constraint because it is only based on relative comparisons and does not have an explicit value, 0 or 1, to give an absolute constraint. Therefore, CL has the problem of insufficient absolute constraints.
2) Many recent contrastive learning methods learn representations with a unit L2 norm constraint has two main properties: alignment and uniformity [23]. Alignment favors the encoder to assign similar features to similar samples. Uniformity prefers to retain the distribution of features with maximum information, that is, the distribution should be uniform. But CL only constrains the distance between the user and the negative items which are only a part of all the items. And a good distribution requires that all items should be distributed as uniformly as possible.

We propose Personalized Contrastive Loss (PCL) to solve the above problems of CL. We give explicit absolute constraints in combination with CE loss to keep the predicted values close to 0 or 1. To address the lack of constraints on the uniformity of the CL, we add more uniformity regularization among samples. The reasonable distance among items facilitates the representations of individual characteristics of the items. Our proposed PCL clearly maintains personalized features and improves the spatial distribution of features relative to CL in Fig. 1.

This article considers personalized sequential recommendations from a fundamental perspective of representation learning, and the quality of embeddings. The proposed loss function works from the perspective of learning objectives and training optimization, is more adaptive, and can be used in many existing models. Therefore, it is of great importance. Moreover, recent studies have shown that self-attention-based sequential recommendation models are inherently subject to the risk of oversmoothing. Self-attention-based sequential recommendations suffer from the same risk of oversmoothing as graph aggregation [24]. Representation degeneration problem is found by recent advancements of sequential models such as Transformer and BERT [16]. Our approach alleviates the problem of insufficient personalized features from a fundamental perspective, i.e., in terms of target constraints. It also theoretically helps to solve the smoothing problem caused by the model. Our contributions are as follows,

- We propose a loss function PCL to solve the new problem that CL makes the feature distribution too compact, which is crucial for personalized feature representation and personalized sequential recommendation.
- We discovery that CL is a relative constraint, so we propose to combine absolute constraints of BCE/CE with it and make them work in a collaborative way.
- We propose to employ more uniformity regularization constraints to compensate for the drawback that CL just make insufficient uniformity constraint.
- Extensive experiments demonstrate the effectiveness of the proposed PCL and achieve state-of-the-art results. The PCL is simple, effective, and adaptive.

II. Related Work

A. Sequential Recommendations

Traditional recommendation systems model user-item interactions in a static manner and can only capture users' general preferences. Sequential Recommendations treat user-item interactions as a dynamic process and take into account the sequential or temporal dependence of the interactions to capture users' current and most recent preferences for more accurate recommendations. Sequential recommendations have become an extensively researched topic because it is more compatible with recommendation scenarios and applications for the following reasons. Both the users' preferences and items' popularity are dynamic over time rather than static [4].

Many techniques and methods are used to make recommendations. As a classical method for sequential recommendation, Markov chains were used for sequential recommendation in the early stage [25], [26]. But they can only capture the short-term dependencies while ignoring long-term ones. Due to the natural strength in sequential modeling, recurrent neural networks (RNNs), such as LSTM, and GRU, are introduced to this field. RNN-based sequential recommendations [27], [28] try to predict the next possible interaction by modeling the sequential dependencies over the given interactions. CNN-based sequential recommendations [29], [30] use the embedding matrix of interaction sequences as an "image" in embedding space, making better use of local features, but failing to make better use of long-term dependencies. As a kind of graph data, graph neural networks (GNNs) are naturally used for sequential recommendation [13], [31], [32], [33], [34], [35]. Complex dependencies in sequences are mined by graph convolution and have the potential to provide interpretability. However, graph-based methods generally have a high computational complexity and training time consumption.

Recently, due to the success of self-attention mechanisms, Self-attention networks(SANs) based or Transformer encoder based methods have been proposed [5], [7], [14], [15]. For example, SASRec [5] uses the self-attention mechanism that can capture long-term semantics while identifying related items and

using them to predict the next item. SASRec yields better performance and is widely used as a baseline for many methods [6], [7], [36]. For example, TiSASRec [6] models both the absolute positions of items as well as the time intervals between them in a sequence based on SASRec. The previous approach uses sequential neural networks to encode the user's historical interactions from left to right as a hidden representation for the recommendation. BERT4Rec [14] uses a deep bidirectional self-attention network to model user behavior. Most sequential recommendation models regard interaction histories as ordered sequences, without regard for the time intervals between each interaction. Contrastive learning is a new and valuable research direction in deep learning, and sequential recommendation methods based on contrastive learning will be stated separately next.

In summary, numerous methods have been proposed to model the sequential recommendation problem. But these are problems of how to represent, little attention has been paid to the quality of the representations. The problem of approximation and smoothness of these representations has been raised [16], [24], which is an important issue that affects the representation of personalized features and the effectiveness of personalized recommendations. Besides, it is clear from the current studies that little attention has been paid to the loss functions in the sequential recommendation. Although BPR [37] is available, Binary cross-entropy (BCE) and cross-entropy(CE) dominate absolutely. We focus on improving the personalized representation problem with a new loss function.

B. Contrastive Learning-Based Sequential Recommendations

Contrastive learning is a kind of self-supervised learning, which has become a hot research topic in recent years for its simplicity and effectiveness [38], [39], [40], [41], [42]. It obtains two different inputs of the same data by data augmentation, and two different transformations of the same data constitute positive pairs and the other data in the same batch constitute negative pairs. Under the constraint of the contrastive loss, the positive pair is pulled to make it close and the negative pair is pushed apart to make it far away. The key is how to augment the data and construct positive and negative pairs.

Many methods based on contrast learning have been proposed and have achieved significant improvements for the sequential recommendation. Existing techniques enhance sequences based on perturbations of random items. Xie et al. [7] propose three data augmentation methods (crop/mask/reorder) to build different views of user interaction sequences to construct self-supervision signals. This contrastive learning-based sequential recommendation method can extract more meaningful user patterns and further encode the user representation effectively. Liu et al. [36] think these methods tend to destroy the original item relationships in the sequence. They proposed two informative augmentation methods, i.e. substitute and insert, which leverage item correlations to generate robust augmented sequences. Qiu et al. [16] find that the distribution of item embeddings generated by recent sequential deep learning models, such as Transformer and BERT, tends to degenerate into an anisotropic shape, which may result in high semantic similarities among embeddings. DuoRec is proposed to improve the item embeddings distribution in two ways in the contrastive learning framework. A contrastive regularization is designed for DuoRec to reshape the distribution of sequence representations and a model-level augmentation is proposed based on Dropout to enable better semantic preserving. Besides, a memory augmented multi-instance contrastive learning method (MMInfoRec) [43] and Intentional Contrastive Learning (ICLRec) [17] which combines contrastive learning and user intent are proposed recently.

Different data augmentations and more ways to construct positive and negative samples are the mainstream research directions for sequential recommendation under the same contrastive learning framework at present. So the contrastive learning-based approaches are similar. We believe that the key to contrastive learning is the contrastive loss function. It utilizes more and better comparisons, which in turn improves the quality of the representation. Therefore, the contrastive loss function-based method is a valuable approach as some related contrastive loss functions [19], [21] have already achieved significant improvements in the top-k recommendation task.

III. Methodology

The sequential recommendation uses historical interactions to infer user's preference and recommend the next item. There is an item set V containing all items and $|V|$ is the number of items. The historical interactions of a user are constructed as an ordered list $\boldsymbol{S_u} = \{\boldsymbol{v_1}, \boldsymbol{v_2}, \ldots, \boldsymbol{v_t}\}$, where $v_i \in V, 0 \leq i \leq t$, and t indicates the current time step as well as the length of s. The recommendation task is to predict the next item v_{t+1} at time step $t+1$ for the user. We propose a loss function PCL, which needs to be combined with a specific method to be applied. We first introduce the baseline model and then present our loss function. As a whole, the method proposed in this article is named PCL4SRec. The method PCL4SRec and PCL are shown in Fig. 2.

A. Baseline Model

Recently, the Transformer encoder is widely adopted for sequential recommendations, and SASRec is widely used as the baseline method as one of the typical models. It uses a self-attention mechanism to automatically learn the user's feature representation at different times from the input user interaction sequences. Generally, there is no embedding of the user in this case, and the user's representation at different time are obtained by feature aggregation of items' embeddings. Here, SASRec or Transformer encoder is used as the baseline model. There are three main layers: Embedding Layer, Transformer encoder, and Prediction Layer.

1) Embedding Layer: Generally, only the embedding of items is available in recently sequential recommendation methods, and the representation of users at different times are obtained from the item embeddings through the model.The input representations of items in the user sequence by adding the item

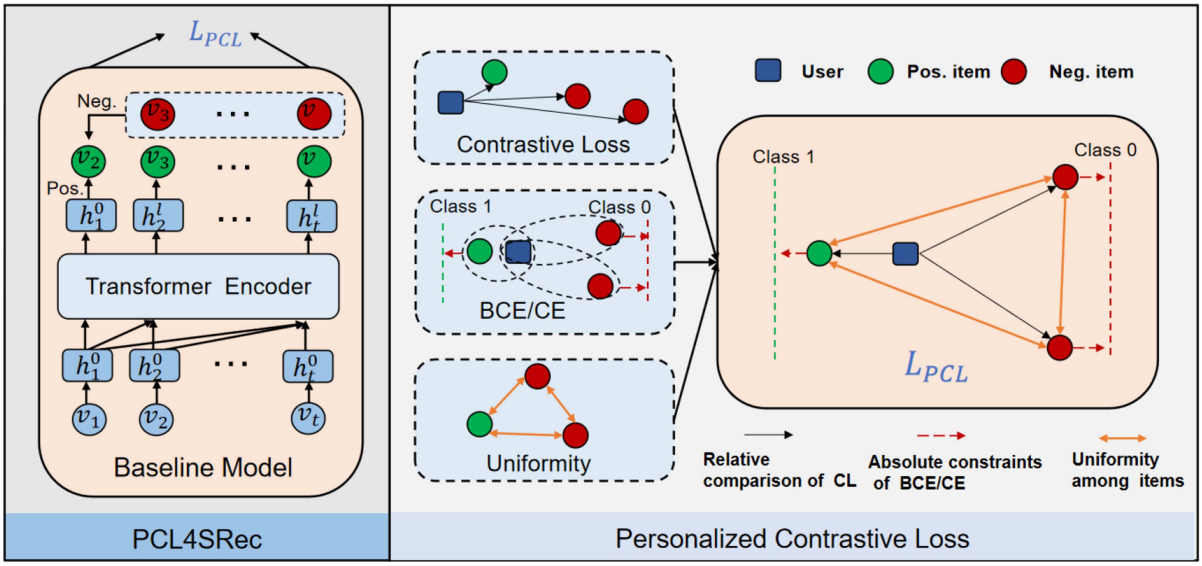


Fig. 2. Proposed method PCL4SRec and PCL. PCL makes the original CL more reasonable distribution of items by BCE/CE loss and Uniformity regularization and thus facilitating the representation of individual characteristics of items.

embedding and position embedding together as:

$$\boldsymbol{h}_i^0 = \boldsymbol{v}_i + \boldsymbol{p}_t \tag{1}$$

2) Transformer Encoder: The self-attention mechanism is used to dynamically capture users' interests at different times, making it an effective way for sequential recommendations. The Transformer encoder adopts a multi-headed self-attention mechanism and position-wise Feed-Forward Network to form the basic block. By stacking these blocks, the Transformer encoder gains powerful feature extraction and transformation capabilities and is widely used. Assuming $\boldsymbol{H^0} = \{\boldsymbol{h_1^0}, \boldsymbol{h_2^0}..., \boldsymbol{h_t^0}\}$ is the initial representation of the sequence, after the L-layer Transformer encoder, a new representation is obtained as:

$$\boldsymbol{H^L} = Trm(\boldsymbol{H^0}) \tag{2}$$

where Trm denotes the Transformer encoder [16], [44]. And the representation of the last layer $\boldsymbol{H}^L$ is used as the final features of the user, and $\boldsymbol{h}_t^L$ is the user representation at timestamp t.

3) Prediction Layer: In the prediction phase, the predicted value is the inner product of the representation of the user at a time and the embedding of the item $\boldsymbol{v}_i$,

$$\hat{y} = \boldsymbol{h_t^L v_i} \tag{3}$$

SASRec optimizes the model using BCE loss, which takes into account the predictions at each time and conducts a negative sampling for each positive item.

B. Personalized Contrastive Loss (PCL)

In this section, we specifically analyze the shortcomings of the CL function and propose improvements to form the new function PCL.

1) Cl: The CL function is widely used, especially in recent contrastive learning where it has played a key role. It can be derived from different perspectives, but all have similar or identical forms [21], [38], [45], [46], [47]. Specifically, CL is defined as [45], [46], [47]:

$$f(u, v) = \boldsymbol{h}_u^\top \boldsymbol{v} / (\|\boldsymbol{h}_u\| \, \|\boldsymbol{v}\|) \tag{4}$$

$$L_{CL} = -\frac{1}{N} \sum \log \frac{\exp(f(u, v^+)/\tau)}{\exp(f(u, v^+)/\tau) + \sum_{v \in V^-} \exp(f(u, v^-)/\tau)} \tag{5}$$

where u, v represents users and items, $\boldsymbol{h_u}, \boldsymbol{v}^+, \boldsymbol{v}^-$ represent the representations of the user, positive item, and negative item, respectively. And V^- is the set of negative samples; N denotes the batch size. $f(u, v)$ is the cosine similarity of the (u, v) pair. The idea of contrastive loss is to pull the user closer to the positive samples and pull the user further away from the negative samples. Generally, non-positive samples within the same batch are used as negative samples, as shown in the top left corner of Fig. 2, and thus can use a large number of negative samples to get excellent results.

CL has two drawbacks which make the embeddings too compact for sequential recommendations. First, it is a relative constraint with no explicit distance. The other problem is that it only widens the space between positive and negative samples. The distribution of ideal embeddings should be uniform that CL has not done yet. PCL's improvements to these two points are shown in Fig. 2.

2) Analysis of Relative and Absolute Constraints: For the first drawback of CL, we analyze it from the perspective of gradient optimization. And we combine it with absolute constraints to bridge this gap.

Relative Constraints of the CL: There are two ways to illustrate this problem.

1. Analysis on the loss function: (5) can be rewritten as follows:

$$L_{CL} = \frac{1}{N}\sum \log\left(1 + \sum_{v\in V^-} \exp((f(u,v^+) - f(u,v^-))/\tau)\right) \tag{6}$$

In the optimization process, the smaller L_{CL} is the better, i.e., the larger $(f(u,v^+) - f(u,v^-)$ is, the better. L_{CL} constrains the distance between $(f(u,v^+)$ and $(f(u,v^-)$ to be as large as possible. So the CL function is a relative constraint because it does not have an explicit distance constraint, value constraint, or decision boundary.

2. Gradient-based approach: Following the previous works [21], [39], we obtain the gradient formulas as follows:

$$\frac{\partial L(u,v)}{\partial h_u} = \frac{1}{\tau \left\|h_u\right\|}\left\{c(v^+) + \sum_{v^-\in V^-} c(v^-)\right\} \tag{7}$$

where

$$\begin{aligned} c(v^+) &= \left(s_{v^+} - \left(s_u^T s_{v^+}\right) s_u\right)^T \left(P_{uv^+} - 1\right), \\ c(v^-) &= \left(s_{v^-} - \left(s_u^T s_{v^-}\right) s_u\right)^T P_{uv^-}, \\ P_{uv} &= \frac{\exp\left(s_u^T s_v/\tau\right)}{\sum_{v\in V}\exp\left(s_u^T s_v/\tau\right)} \end{aligned} \tag{8}$$

where s_u, s_v are the normalized representations, i.e., $s_u = \boldsymbol{h_u}/\|\boldsymbol{h_u}\|, s_v = \boldsymbol{v}/\|\boldsymbol{v}\|$.

Equation (7) consists of two components, $c(v^+), c(v^-)$. Equation 8 shows that the gradient of positive items, $c(v^+)$, is related to the distance between P_{uv^-} and 1, and the gradient of negative items, $c(v^-)$, is related to the distance between P_{uv^-} and 0. However, the key factor P_{uv} is a ratio, which is indeed a relative value gained by the softmax function. In conclusion, the results of the gradient formulas clearly show that the CL is influenced by the factor P_{uv} in the optimization process. It is a relative comparison process.

Absolute Constraints of BCE and CE: This shortcoming of the CL can be compensated by the traditional BCE or CE loss functions. They are widely adopted loss functions, especially in the field of classification. BCE and CE are also the mostly used loss functions in the field of sequential recommendation to recent articles. They are constrained by explicit one-hot vectors, 0 or 1, to form different classification boundaries. Their formulas are as follows,

$$L_{BCE} = -\left[y_i \log\left(\sigma\left(\hat{y}_i\right)\right) + \left(1 - y_i\right)\log\left(1 - \sigma\left(\hat{y}_i\right)\right)\right] \tag{9}$$

$$L_{CE} = -\sum_{i=1}^{N} y_i \log\left(\sigma\left(\hat{y}_i\right)\right) \tag{10}$$

where $\hat{y}_i$ is the predicted value and y_i is the true value of one-hot data. σ is the activation function, sigmoid is used in BCE, and softmax is used in CE. Their derivatives are as follows,

$$\frac{dL_{BCE}}{d\hat{y}_i} = \sigma\left(\hat{y}_i\right) - y_i \tag{11}$$

$$\frac{dL_{CE}}{d\hat{y}_i} = y_i\left(\sigma\left(\hat{y}_i\right) - 1\right) \tag{12}$$

It can be seen that BCE constrains both positive and negative items close to 1 or 0. CE is used for multi-classification tasks and makes the probability of class i close to 1, i.e., close to its corresponding decision surface when $y_i = 1$. Thus, both BCE and CE are absolute constraints and have explicit classification boundaries.

3) Making Embeddings Distribution More Uniform: All the embeddings should be uniformly distributed but CL just makes the positive and negative items far away from each other. We add uniformity regularization for this.

Uniformity of the CL: Recently, Wang et al. [23] argue that uniformity prefers a feature distribution that preserves maximal information, i.e., the uniform distribution on the unit hypersphere. Contrastive learning normalizes their features on the unit hypersphere, and the unit hypersphere is indeed a nice feature space. It is crucial that uniformity of feature distributions on the output unit hypersphere. They also prove that the CL optimizes this property asymptotically. The uniformity is defined as:

$$L_{uniform} = \log \underset{x,y\sim p_{data}}{E}\left[e^{-2\|f(x)-f(y)\|_2^2}\right] \tag{13}$$

where p_{data} is the distribution of the independent samples, x, y are two samples in the data, $f(x), f(y)$ are the representations of them. Minimizing $L_{uniform}$ is equivalent to encouraging the uniform distribution of the representations of all the data samples from the distribution p_{data}.

Adding more Uniformity Regularization: The uniformity constraint of the CL function is insufficient for it just pushes those negative items and makes them uniformly distributed. A good embedding representation requires all items uniformly distributed, not just between positive and negative samples. Therefore, we add uniformity among positive items, and among negative items as a regularization constraint on the data distribution. Therefore, the following equation is used in this article,

$$\begin{aligned} L_{uniform} = &\log \underset{v_1,v_2\sim V^-}{E}\left[e^{-2\|f(v_1)-f(v_2)\|_2^2}\right] \\ &+ \log \underset{v_3,v_4\sim V^+}{E}\left[e^{-2\|f(v_3)-f(v_4)\|_2^2}\right] \end{aligned} \tag{14}$$

where v_1, v_2, v_3, v_4 are items, V^-, V^+ are negative and positive item sets, respectively.

Existing sequential recommendations tend to model local relationships. For example, the relationship between items or the relationship between users and items. Even if relationships within the whole sequence are considered, this is a localized relationship within a user unit, although it is a larger scope than the former. Here the users or items are required to be evenly distributed overall, which is a global constraint that compensates for the shortcomings of the previous local modeling.

4) Combining Absolute and Uniformity Constraints as PCL: Based on the above detailed analysis, we use BCE or CE to add absolute constraints to CL to make positive and negative samples closer to the decision boundary of the classification. This is also multi-task learning in different constraint spaces, which helps to learn better feature representations. We enhance

TABLE I
STATISTICS OF THE DATASETS

Dataset	Beauty	Sports	Toys	ML-1M
Users	22 363	25 598	19 412	6 040
Items	12 101	18 357	11 924	3 953
Avg.Length	8.9	8.3	8.6	165.6
Actions	198 502	296 337	167 597	1 000 209
Sparsity	99.93 %	99.95 %	99.93 %	95.81 %

the regularization constraint on the uniform distribution of items in the feature space to make their distribution more reasonable.

$$L_{PCL} = \alpha * L_{CL} + (1-\alpha) * L_{BCE} + \lambda * L_{uniform} \quad (15)$$

or

$$L_{PCL} = \alpha * L_{CL} + (1-\alpha) * L_{CE} + \lambda * L_{uniform} \quad (16)$$

where α is the weight of the CL and $\alpha \in [0, 1]$, which makes the two losses work in a collaborative and balanced way; λ is the regularization parameter. L_{BCE} is used for SASRec and it can be replaced by L_{CE} as needed in other models.

L_{BCE} / L_{CE} and $L_{uniform}$ widen the nodes distance from different perspectives. They solve the problem of smoothing and insufficient personalization due to feature approximation and distribution concentration. This allows for better feature representation, especially for enhanced personalized features.

IV. EXPERIMENTS

Experiments are conducted to verify the effectiveness of the PCL4SRec and the PCL function. The experimental setup, results, ablation study, and discussion of hyperparameters are presented in detail in this section. We also show the adaptability of PCL and give visual and quantitative analyses related to personalization.

A. Experimental Setup

1) Datasets: We conduct experiments on 5 commonly used datasets collected from real-world platforms in different domains and sparsity levels. We use three subcategories: "Beauty", "Sports and Outdoors", and "Toys and Games" from the Amazon reviews dataset, and they are abbreviated as "Beauty", "Sports, and "Toys", respectively [48]. The ML-1 M (MovieLens 1 M) dataset is used for it is one of the most classic datasets [49]. We group the interaction records by the user and sort them in ascending order by interaction timestamp. And the statistics of these datasets are summarized in Table I.

2) Evaluation Metrics: To avoid the biased discoveries problems associated with the sampled evaluation approach [50], we use all the items for evaluation without negative sampling [7], [15], [16], [36]. For overall evaluation, top-K Hit Ratio (HR@K) and Normalized Discounted Cumulative Gain(NDCG@ K) are applied with $K = 5, 10$. HR focuses on the presence of positive terms, while NDCG further considers the quality of the ranking.

3) Baseline Methods: The most traditional method POP is used, which is also a non-sequential recommendation method. Deep learning based methods, such as GRU4Rec [27], NARM [51], SASRec [5], CoSeRec [36], LightSANs [15], DuoRec [16] and SP-PLR [52], are used for comparison. They employ advanced techniques such as RNN, attention mechanism, Transformer, contrastive learning, and so on. The latter three are competitive methods that have been proposed recently.

4) Implementation Details: We adopt the leave-one-out strategy to evaluate the performance of each method, which is widely employed in many related works [5], [16], [48]. For each user, we use the last two interactions as validation data and test data, respectively, and the other items before them are used for training. For a fair comparison, as a regular setting following previous works [7], [16], [36], the embedding dimension size is 64, the batch size is set to 256, and the maximum sequence length is set to 50. For the baseline model of SASRec, we stack 2 self-attention blocks together and set the head number as 2 for each block. We use Adam optimizer to optimize the parameters. We train the model with early stopping techniques according to the performance on the validation set.

B. Performance Comparison

Table II shows the experimental results where the best results are bolded and the underlined ones are the second-best results. The differences under random seeds are statistically small, for example, the standard deviations of NDCG@10 on the four datasets are $\pm 0.0006, \pm 0.0004, \pm 0.0003, \pm 0.001$, respectively. The proposed PCL4SRec achieves the best results on all datasets and metrics.

POP recommends directly according to the most popular items without considering the characteristics of any users and items. The results are the worst among these methods. GRU4Rec takes advantage of the RNN to model the historical sequences and performs better than POP. NARM models the user's purpose using the attention mechanism based on GRU and outperforms GRU4Rec overall which shows the effectiveness of modeling the user's purpose. SASRec is a self-attention-based method that can mine the sequence and its dependencies to automatically calculate the weights of each item. A significant and consistent improvement over previous methods is obtained. This shows the advantage of the self-attention mechanism for sequential recommendations. And the methods based on the self-attentive mechanism, Transformer, have become the baseline for many subsequent methods. CoSeRec is a recommendation method based on contrastive learning, with significant performance improvement on Beauty and Sports datasets. Contrastive learning methods made better use of data structures through data augmentation and learn by multi-tasking. But CoSeRec introduces more parameters, while the parameter tuning of our method is simple.

LightSANs, DuoRec and SR-PLR [52] are the state-of-the-art (SOTA) methods proposed recently. DuoRec is also in a contrastive learning framework, and PCL4SRec directly constrains the feature distribution of the items and gains better results. SR-PLR is the the latest method which combins deep learning and symbolic learning in a dual feature-logic network, and modeling users'dynamic preferences with a probabilistic method. Table II shows that our method is superior to SR-PLR, mainly

TABLE II
PERFORMANCE COMPARISON

Datasets	Metric	POP	GRU4Rec	NARM	SASRec	CoSeRec	LightSANs	DuoRec	SR-PLR	PCL4SRec	Improv. v.s. SASRec	Improv. v.s. All
Beauty	HR@5	0.0074	0.0212	0.0257	0.0415	0.0537	0.0580	0.0546	0.0545	**0.0644**	55.18%	11.03%
	HR@10	0.0118	0.0370	0.0464	0.0609	0.0752	0.0872	0.0845	0.0826	**0.0926**	52.05%	6.19%
	NDCG@5	0.0041	0.0137	0.0158	0.0264	0.0361	0.0356	0.0352	0.0333	**0.0440**	66.67%	21.88%
	NDCG@10	0.0056	0.0187	0.0224	0.0326	0.0430	0.0450	0.0443	0.0424	**0.0530**	62.58%	17.78%
Sports	HR@5	0.0057	0.0103	0.0107	0.0202	0.0287	0.0340	0.0326	0.0332	**0.0383**	89.60%	12.65%
	HR@10	0.0091	0.0179	0.0192	0.0322	0.0437	0.0538	0.0498	0.0515	**0.0555**	72.36%	3.16%
	NDCG@5	0.0041	0.0062	0.0065	0.0135	0.0196	0.0196	0.0208	0.0192	**0.0264**	95.56%	26.92%
	NDCG@10	0.0052	0.0086	0.0092	0.0174	0.0242	0.0260	0.0262	0.0252	**0.0319**	83.33%	21.76%
Toys	HR@5	0.0056	0.0217	0.0229	0.0491	0.0445	0.0646	0.0665	0.0632	**0.0691**	40.73%	3.91%
	HR@10	0.0089	0.0343	0.0401	0.0707	0.0658	0.0946	0.0950	0.0919	**0.0977**	38.19%	2.84%
	NDCG@5	0.0037	0.0140	0.0141	0.0341	0.0296	0.0386	0.0396	0.0359	**0.0495**	45.16%	25.00%
	NDCG@10	0.0048	0.0181	0.0195	0.0411	0.0365	0.0484	0.0487	0.0441	**0.0587**	42.82%	20.53%
ML-1M	HR@5	0.0209	0.1055	0.1050	0.1364	0.1227	0.1522	0.1500	0.1384	**0.1810**	32.70%	18.92%
	HR@10	0.0368	0.1760	0.1829	0.2176	0.1975	0.2359	0.2336	0.2262	**0.2631**	20.91%	11.53%
	NDCG@5	0.0129	0.0644	0.0624	0.0899	0.0800	0.0966	0.0957	0.0857	**0.1208**	34.37%	25.05%
	NDCG@10	0.0180	0.0871	0.0874	0.1160	0.1042	0.1237	0.1227	0.1139	**0.1472**	26.90%	19.00%

The bolded results in the table are the best results and the underlined ones are the second-best results. The improvement relative to the baseline SASRec and the second-best results are listed on the right.

TABLE III
ABLATION STUDY

SASRec	CL	BCE	Uniform	Beauty		Sports		Toys		ML-1M	
				HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
✓		✓		0.0609	0.0326	0.0322	0.0174	0.0707	0.0411	0.2176	0.1160
✓	✓			0.0812	0.0449	0.0431	0.0245	0.0900	0.0518	0.2233	0.1200
✓	✓	✓		0.0877	0.0500	0.0530	0.0298	0.0967	0.0568	0.2558	0.1398
✓	✓		✓	0.0901	0.0516	0.0531	0.0306	0.0912	0.0527	0.2558	0.1412
✓		✓	✓	0.0626	0.0335	0.0361	0.0200	0.0728	0.0419	0.2214	0.1176
✓	✓	✓	✓	**0.0926**	**0.0530**	**0.0555**	**0.0319**	**0.0977**	**0.0587**	**0.2631**	**0.1472**
improv. v.s. BCE				52.05%	62.58%	72.36%	83.33%	38.19%	42.82%	20.91%	26.90%
improv. v.s. CL				14.04%	18.18%	28.88%	30.41%	8.47%	13.20%	17.79%	22.71%

because we introduce and improve the contrastive loss in sequential recommendation, which addresses the personalization problem from two specific aspects. Our approach is more adaptable to sequential recommendation scenarios and more effective. What more, experiments show that this latest method has worse performance than the first two methods, LightSANs and DuoRec. These two approaches both perform well overall but differ on different datasets. As shown in the last two columns of the table, the improvements of PCL4SRec relative to the baseline method SASRec are very significant, with the lowest being over 20% and the highest being nearly 96%. Relative to the second-best results, our method boosts mostly above ten percent, with a maximum boost of nearly 27%. Importantly, the improvement obtained from the loss function is interpretable and it is also model-independent, which can be verified subsequently.

C. Ablation Study

The results of the ablation analysis are in Table III. The first row is the BCE-based methods, which are the worst among all methods. The second row is the CL-based methods, which are better than BPR methods and show the advantage of the CL function. The combination of the two in the third row yields better results than one of them, showing the effectiveness of combining BCE and CL. The fourth row is CL with Uniform loss, and the fifth row is BCE with Uniform loss. As can be seen in the above table, after adding Uniform loss, the performances of both BCE and CL are improved, but not as good as that of PCL. And CL with Uniform loss is much better than BCE with Uniform loss. The best result is our PCL in the sixth row. All these comparisons demonstrate the effectiveness of the method proposed in this article.

The improvements of PCL relative to BCE and CL are listed at the bottom of the table. The results show that PCL has a significant improvement relative to BCE, and also has a consistent improvement relative to CL, but is smaller than BCE because CL itself outperforms BCE. Overall, the last two rows of the table show improved performance with the addition of the new factors, and the PCL is optimal. Also, we observe that in both NDCG boosts are higher than HR, which is consistent with previous studies that CL is more helpful in boosting ranking metrics. [19], [21].

D. Adaptability of PCL

To verify the adaptability and effectiveness of PCL, we chose three datasets Beauty, Toys and ML-1 M and five methods, NPE [53], NARM [51], GRU4Rec [27], LightSANs [15], and FMLP-Rec [54] for experiments. Last two are strong baselines

TABLE IV
ADAPTABILITY AND EFFECTIVENESS OF PCL

Model		Beauty		Toys		ML-1M	
		HR@10	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10
NPE	CL	0.0232	0.0104	0.0258	0.0141	0.0202	0.0088
	PCL	0.0503	0.0236	0.0584	0.0272	0.0459	0.0220
NARM	CL	0.0706	0.0381	0.0633	0.0327	0.1551	0.0726
	PCL	0.0715	0.0395	0.0773	0.0460	0.2210	0.1123
GRU4Rec	CL	0.0548	0.0272	0.0640	0.0330	0.1497	0.0727
	PCL	0.0751	0.0423	0.0773	0.0462	0.2310	0.1192
LightSANs	CL	0.0845	0.0425	0.0915	0.0452	0.2111	0.1063
	PCL	0.0901	0.0459	0.1016	0.0496	0.2253	0.1138
FMLP-Rec	CL	0.0805	0.0441	0.0912	0.0527	0.2219	0.1137
	PCL	0.0930	0.0535	0.0996	0.0596	0.2616	0.1409
SASRec	CL	0.0812	0.0449	0.0900	0.0518	0.2233	0.1200
	PCL	0.0926	0.0530	0.0977	0.0587	0.2631	0.1472

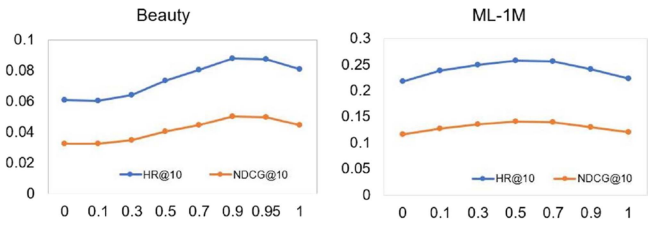


Fig. 3. Impact of the weight of CL α.

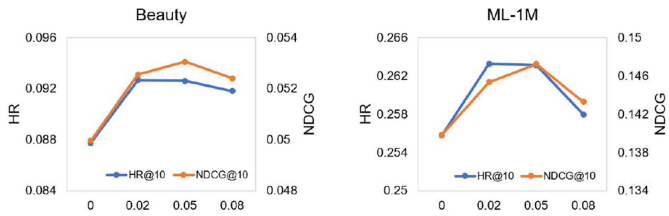


Fig. 4. Impact of the weight of Uniformity λ.

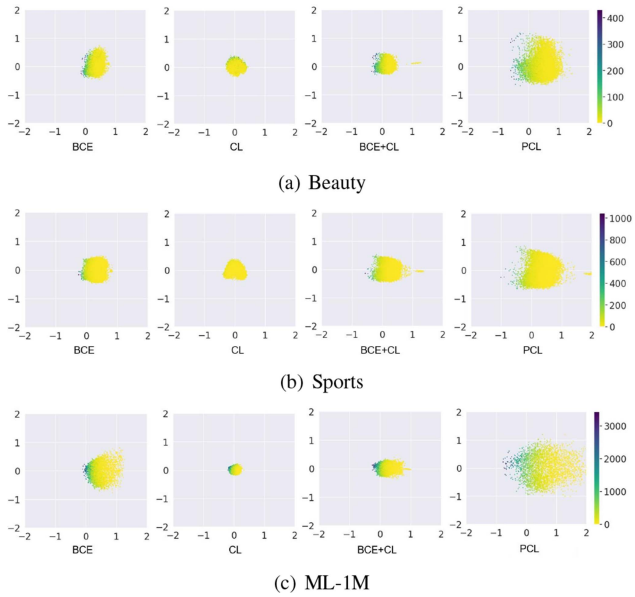
proposed recently. The baseline of this article, SASRec, is placed at the end for the sake of comparison. Results are shown in Table IV.

Compared with CL-based methods, PCL-based methods show better performances in different models, and most of the comparisons show a substantial improvement with respect to CL. The performance on the new methods, LightSANs and FMLP-Rec, are better than the other methods. Overall, FMLP-Rec achieves the best performance. However, it is worth noting that FMLP-Rec has a similar performance overall compared to SASRec which shows that SASRec is still a strong baseline as is widely used by many other works. Overall, PCL-based methods perform consistently better than CL-based methods on different datasets and different methods, which illustrates that PCL has good adaptability and effectiveness.

E. Hyperparametric Discussion

1) Impact of the Weight of CL α: α is the weight of CL, which makes the balance between CL and BCE. Fig. 3 shows that generally the performance improves as the weight of CL increases, and decreases after reaching the peak. In general, CL requires a larger α, such as 0.9, 0.95, etc, which indicates that CL can play an important role compared to BCE. PCL becomes BCE or CL loss when $\alpha = 0$ or $\alpha = 1$. PCL obtains the maximum value when $\alpha \in (0, 1)$, which is consistent with the ablation analysis.

2) Impact of the Weight of Uniformity λ: Fig. 4 shows the HR@10 and NDCG@10 on different λ. Different datasets have different performances but share similar trends. Both values in Fig. 4 rise and then fall both on Beauty and ML-1 M, while the latter declines more rapidly. $\lambda = 0$ on the left are the cases without this constraint and their performances are generally the worst. This also demonstrates the effectiveness of uniformity regularization.



(a) Beauty

(b) Sports

(c) ML-1M

Fig. 5. Visualization Analysis. Colors indicate the frequency of items. PCL owns the largest feature spaces which facilitate the representation of individual characteristics.

TABLE V
PCL COMPARED WITH MANY OTHER LOSS FUNCTIONS IN DIFFERENT NEGATIVE SAMPLES

Dataset	Neg.Sample	Loss	NARM		GRU4Rec		LightSANs	
			HR	NDCG	HR	NDCG	HR	NDCG
Beauty	one	BPR	0.0464	0.0224	0.0369	0.0186	0.0392	0.0174
		BCE	0.0357	0.0168	0.0313	0.0151	0.0436	0.0183
		CL	0.0336	0.0162	0.0472	0.0232	0.0696	0.0337
		PCL	**0.0720**	0.0389	0.0673	0.0342	0.0708	0.0347
	all	CE	0.0717	0.0377	0.0632	0.0347	0.0879	0.0450
		CL	0.0706	0.0381	0.0548	0.0272	0.0845	0.0425
		PCL	0.0715	**0.0395**	**0.0751**	**0.0423**	**0.0901**	**0.0459**
Toys	one	BPR	0.0401	0.0195	0.0343	0.0181	0.0413	0.0187
		BCE	0.0444	0.0227	0.0339	0.0171	0.0407	0.0172
		CL	0.0643	0.0315	0.0585	0.0297	0.0792	0.0375
		PCL	**0.0780**	0.0413	0.0771	0.0409	0.0785	0.0388
	all	CE	0.0574	0.0310	0.0547	0.0300	0.0946	0.0484
		CL	0.0633	0.0327	0.0640	0.0330	0.0915	0.0452
		PCL	0.0773	**0.0460**	**0.0773**	**0.0462**	**0.1016**	**0.0496**

The "Neg. Sample" in the table is short for negative samples.

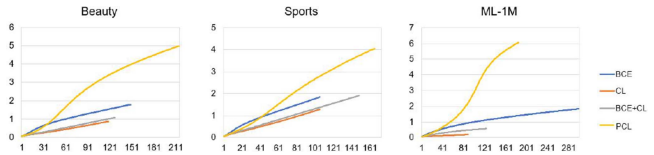


Fig. 6. Variance of all items of Beauty and ML-1 M datasets with different training epochs. PCL has a significantly large variance and measures the degree of individualization overall.

F. Personalization Analysis

Personalization is studied using different loss functions under the same baseline SASRec.

1) Visualization Analysis: Fig. 5 shows the spatial distribution of latent vectors for Beauty and ML-1 M under four constraints, respectively. Following DuoRec [16], the item embedding matrix of the dataset is projected into 2D by SVD with colors indicating the frequency of items in the dataset.

Results show that CL is the most compact and BCE is a little larger relative to CL. The feature space of BCE+CL is larger than CL. That of PCL is the largest, and the distribution of PCL is the most uniform among the four. There is a significant spatial expansion of PCL relative to BCE+CL which indicates the effectiveness of uniform regularization. It can be noticed that some outliers on the right side of BCE+CL and PCL can be seen in these figures as a small region. When considering this, they have a larger feature space. The formation of these small regions may be because CL and BCE are different kinds and spaces of constraints, and different dominant factors form these two regions.

2) Quantitative Analysis: The variance represents the degree of dispersion of the overall distribution and provides an overall measure of the personalization of all items [22]. Fig. 6 shows the changes in variance during the overall training for the four different loss functions. The horizontal axis is the number of training epochs.

Results show that CL generally converges the fastest but has the smallest variance. The variance of BCE is larger than that of CL, and the variance of BCE+CL is in the middle of the two. These are consistent with the above visualization analysis. However, this figure clearly shows that the variance of BCE+CL is between CL and BCE, where BCE converges slower than CL. The variance of PCL is the largest at the last of the training. A slow increase in the variance of PCL at the beginning of the training is observed in the figure, followed by a steep increase.These results show the difference in the role of the three losses at different stages. Expanding the feature space and increasing personalization in the middle and late stages of training become the main goals of PCL training. The good performance of PCL in enhancing personalized representation is shown quantitatively by Fig. 6.

G. PCL and Other Loss Functions

To compare the differences between PCL and other loss functions we choose two of the datasets and three methods, NARM, GRU4Rec and LightSANs, to study the performance of PCL. NARM constrains the uniform distribution of item and user features, while the proposed PCL4SRec and GRU4Rec with PCL constrain only item features. In addition, we consider the CE loss, which uses all items and maximizes the probability of positive samples. Thus, we divide into two modes depending on the number of negative samples used, such as "one" and "all" in Table V, and the corresponding CL uses the same number of negative samples. When sampling one negative sample, we use BPR, BCE, CL and PCL, and when using all items we use CE, CL, and PCL. A comprehensive consideration of various loss functions is given here, and the results are in Table V. There are three following observations:

1. In four group comparisons of the two datasets, PCL was the best in each group. BPR and BCE perform differently in different datasets. Generally, CL loss is better than CE/BCE. In addition, the last three lines of each group of CE or BCE, CL, and PCL can confirm the effectiveness of our PCL.

2. The comparison of sampling methods, i.e., the comparison between the two groups of "one" and "all". CE on all samples is significantly better than BCE based on a single negative sample. CL and PCL on all samples generally tend to achieve better results as well. Overall, all sample-based losses outperform the one sample-based losses, but there are differences across datasets.
3. Overall, the seven results on the two datasets show that the general PCL achieves optimal or suboptimal results, and the PCL based on all samples performs better, except for the performance of LightSANs with CE loss in "all" mode. Our proposed PCL is better than all other loss functions. For the recent model LightSANs, PCL also achieves better results than BCE/CE and CL overall, but the relative improvement is smaller than the first two models. Because the improvement is more difficult for the better model. Therefore, the effectiveness of our proposed PCL can also be illustrated on the advanced baseline model.

In conclusion, PCL is optimal among various loss functions and for different numbers of samples. All samples based loss function tends to achieve better results.

V. Conclusion

We find the problem that CL makes the feature space too compact in sequential recommendations, so CL function is unfriendly for the personalized sequential recommendation because it is a relative constraint and has insufficient uniformity constraints on users and items. The proposed PCL reasonably expands the feature space and makes its distribution more uniform with an absolute constraint and more uniform regularization. This facilitates the representation of personalized features, which in turn significantly improves the recommendation performance. The personalized features are measured qualitatively and quantitatively by two methods. Experiments and analyses verified the state-of-the-art performance and adaptability of PCL, despite its simplicity.

This article shows the importance of the loss function for recommendation systems. Unlike recommendation methods based on contrastive learning frameworks, the method in this article extends the application of contrastive learning and CL function. The spatial distribution of features and the quality of embeddings are also noteworthy research directions for sequential recommendations, and for other tasks of recommendation systems.

References

[1] R. Chen et al., "A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks," *IEEE Access*, vol. 6, pp. 64301–64320, 2018.

[2] Y. Wu, K. Li, G. Zhao, and X. Qian, "Personalized long- and short-term preference learning for next POI recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 4, pp. 1944–1957, Apr. 2022.

[3] Y. Wu et al., "State graph reasoning for multimodal conversational recommendation," *IEEE Trans. Multimedia*, early access, Mar. 03, 2022, doi: 10.1109/TMM.2022.3155900.

[4] S. Wang et al., "Sequential recommender systems: Challenges, progress and prospects," in *Proc. Int. Joint Conf. Artif. Intell.*, 2019, pp. 6332–6338.

[5] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *Proc. IEEE Int. Conf. Data Mining*, 2018, pp. 197–206.

[6] J. Li, Y. Wang, and J. McAuley, "Time interval aware self-attention for sequential recommendation," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2020, pp. 322–330.

[7] X. Xie et al., "Contrastive learning for sequential recommendation," in *Proc. IEEE Int. Conf. Data Eng.*, 2022, pp. 1259–1273.

[8] D. Wang et al., "Modeling sequential listening behaviors with attentive temporal point process for next and next new music recommendation," *IEEE Trans. Multimedia*, vol. 24, pp. 4170–4182, 2022.

[9] Y. Ding, Y. Ma, Wai Keung Wong, and T-S Chua, "Modeling instant user intent and content-level transition for sequential fashion recommendation," *IEEE Trans. Multimedia*, vol. 24, pp. 2687–2700, 2022.

[10] J. Hao, Y. Dun, G. Zhao, Y. Wu, and X. Qian, "Annular-graph attention model for personalized sequential recommendation," *IEEE Trans. Multimedia*, vol. 24, pp. 3381–3391, 2022.

[11] G. Chen, X. Zhang, Y. Zhao, C. Xue, and J. Xiang, "Exploring periodicity and interactivity in multi-interest framework for sequential recommendation," in *Proc. Int. Joint Conf. Artif. Intell.*, 2021, pp. 1426–1433.

[12] Z. Xie et al., "Adversarial and contrastive variational autoencoder for sequential recommendation," in *Proc. Web Conf.*, 2021, pp. 449–459.

[13] Z. Fan et al., "Continuous-time sequential recommendation with temporal graph collaborative transformer," in *Proc. 30th ACM Int. Conf. Inf. & Knowl. Manage.*, 2021, pp. 433–442.

[14] F. Sun et al., "BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer," in *Proc. 28th ACM Int. Conf. Inf. & Knowl. Manage.*, 2019, pp. 1441–1450.

[15] X. Fan et al., "Lighter and better: Low-rank decomposed self-attention networks for next-item recommendation," in *Proc. 44th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2021, pp. 1733–1737.

[16] R. Qiu, Z. Huang, H. Yin, and Z. Wang, "Contrastive learning for representation degeneration problem in sequential recommendation," in *Proc. 15th ACM Int. Conf. Web Search & Data Mining*, 2022, pp. 813–823.

[17] Y. Chen et al., "Intent contrastive learning for sequential recommendation," in *Proc. ACM Web Conf.*, 2022, pp. 2172–2182.

[18] X. Tong, P. Wang, C. Li, L. Xia, and S.Z. Niu, "Pattern-enhanced contrastive policy learning network for sequential recommendation," in *Proc. 13th Int. Joint Conf. Art. Intell.*, 2021, pp. 1593–1599.

[19] H. Tang, G. Zhao, Y. Wu, and X. Qian, "Multisample-based contrastive loss for top-k recommendation," *IEEE Trans. Multimedia*, vol. 25, pp. 339–351, 2023.

[20] H. Tang, G. Zhao, Y. He, Y. Wu, and X. Qian, "Ranking-based contrastive loss for recommendation systems," *Knowl. Based Syst.*, vol. 261, 2023, Art. no. 110180.

[21] J. Wu et al., "On the effectiveness of sampled softmax loss for item recommendation," *CoRR*, 2022, *arXiv:2201.02327*.

[22] L. Chen, L. Wu, R. Hong, K. Zhang, and M. Wang, "Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 27–34.

[23] T. Wang and P. Isola, "Understanding contrastive representation learning through alignment and uniformity on the hypersphere," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 9929–9939.

[24] H. Shi et al., "Revisiting over-smoothing in BERT from the perspective of graph," in *Proc. Int. Conf. Learn. Representations*, 2022.

[25] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 811–820.

[26] R. He and J. McAuley, "Fusing similarity models with markov chains for sparse sequential recommendation," in *Proc. IEEE 16th Int. Conf. Data Mining*, 2016, pp. 191–200.

[27] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," in *Proc. Int. Conf. Learn. Representations*, 2016.

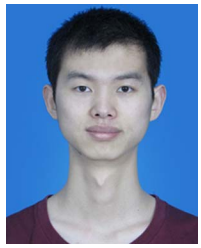
[28] B. Hidasi, M. Quadrana, A. Karatzoglou, and D. Tikk, "Parallel recurrent neural network architectures for feature-rich session-based recommendations," in *Proc. 10th ACM Conf. Recommender Syst.*, 2016, pp. 241–248.

[29] J. Tang and K. Wang, "Personalized Top-N sequential recommendation via convolutional sequence embedding," in *Proc. 11th ACM Int. Conf. Web Search Data Mining*, 2018, pp. 565–573.

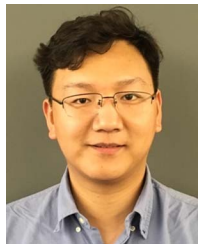
[30] F. Yuan, A. Karatzoglou, I. Arapakis, J. M. Jose, and X. He, "A simple convolutional generative network for next item recommendation," in *Proc. 12th ACM Int. Conf. Web Search & Data Mining*, 2019, pp. 582–590.

[31] J. Chang et al., "Sequential recommendation with graph neural networks," in *Proc. 44th Int. ACM SIGIR Conf. Res. & Develop. Inf. Retrieval*, 2021, pp. 378–387.

[32] Y. Wei et al., "MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video," in *Proc. ACM Multimedia*, 2019, pp. 1437–1445.
[33] C. Ma et al., "Memory augmented graph neural networks for sequential recommendation," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 5045–5052.
[34] Y. Wei et al., "Hierarchical user intent graph network for multimedia recommendation," *IEEE Trans. Multimedia*, vol. 24, pp. 2701–2712, 2022.
[35] H. Liu et al., "Hamming spatial graph convolutional networks for recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 24, pp. 2701–2712, 2022.
[36] Z. Liu et al., "Contrastive self-supervised sequential recommendation with robust augmentation," *CoRR*, 2021, *arXiv:2108.06479*.
[37] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proc. 25th Conf. Uncertainty Art. Intell.*, 2009, pp. 452–461.
[38] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
[39] P. Khosla et al., "Supervised contrastive learning," in *Proc. Int. Adv. Conf. Neural Inf. Process. Syst.*, 2020.
[40] X. Song and Z. Jin, "Robust label rectifying with consistent contrastive-learning for domain adaptive person re-identification," *IEEE Trans. Multimedia*, vol. 24, pp. 3229–3239, 2022.
[41] S. Chang, Y. Li, S. Shen, J. Feng, and Z. Zhou, "Contrastive attention for video anomaly detection," *IEEE Trans. Multimedia*, vol. 24, pp. 4067–4076, 2022.
[42] X. Huo et al., "Heterogeneous contrastive learning: Encoding spatial information for compact visual representations," *IEEE Trans. Multimedia.*, vol. 24, pp. 4224–4235, 2022.
[43] R. Qiu, Z. Huang, and H. Yin, "Memory augmented multi-instance contrastive predictive coding for sequential recommendation," in *Proc. IEEE Int. Conf. Data Mining Workshops*, 2021, pp. 519–528.
[44] A. Vaswani et al., "Attention is all you need," in *Proc. Int. Adv. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[45] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Proc. Int. Adv. Conf. Neural Inf. Process. Syst.*, 2016, pp. 1857–1865.
[46] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *CoRR*, 2018, *arXiv:1807.03748*.
[47] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3733–3742.
[48] K. Zhou et al., "S3-Rec: Self-supervised learning for sequential recommendation with mutual information maximization," in *Proc. 29th ACM Int. Conf. Inf. & Knowl. Manage.*, 2020, pp. 1893–1902.
[49] F. M. Harper and J. A. Konstan, "The Movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 19:1–19:19, 2016.
[50] W. Krichene and S. Rendle, "On sampled metrics for item recommendation," in *Proc.26th ACM SIGKDD Int. Conf. KDD*, 2020, pp. 1748–1757.
[51] J. Li et al., "Neural attentive session-based recommendation," in *Proc. ACM Conf. Inf. & Knowl. Manage.*, 2017, pp. 1419–1428.
[52] H. Yuan et al., "Sequential recommendation with probabilistic logical reasoning," Apr. 2023, *arXiv:2304.11383*.
[53] T. Nguyen and A. Takasu, "NPE: Neural personalized embedding for collaborative filtering," in *Proc. Int. Joint Conf. Artif. Intell.*, 2018, pp. 1583–1589.
[54] K. Zhou, H. Yu, W. X. Zhao, and J-R Wen, "Filter-enhanced MLP is all you need for sequential recommendation," in *Proc. ACM Web Conf.*, 2022, pp. 2388–2399.



Hao Tang received the B.E. degree from Information Engineering University, Zhengzhou, China, M.E. degree from Shandong University of Science and Technology, Qingdao, China, in 2011 and 2013, respectively, and the Ph.D. degree from Xi'an Jiaotong University, Xi'an, China in 2022. He is currently with China Unicom Shaanxi Branch. His research interests include recommendation systems, graph neural networks, and the contrastive learning.



Guoshuai Zhao (Member, IEEE) received the B.E. degree from Heilongjiang University, Harbin, China, in 2012, the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 2015 and 2019, respectively. He was an intern with the Social Computing Group, Microsoft Research Asia in 2017, a Visiting Scholar with Northeastern University, Boston, MA, USA, from 2017 to 2018 and MIT, Cambridge, MA, USA, in 2019. He is currently an Associate Professor with Xi'an Jiaotong University. His research interests include social media Big Data analysis, recommendation systems, and natural language generation.

Jing Gao received the B.E. degree from Xi'an University of Posts & Telecommunications, Xi'an, China, in 2016. After working for one year, he is currently working toward the M.E. degree with SMILES LAB, Xi'an Jiaotong University, Xi'an. His research focuses on recommendation systems.

Xueming Qian (Member, IEEE) received the B.S. and M.S. degrees from the Xi'an University of Technology, Xi'an, China, in 1999 and 2004, respectively, and the Ph.D. degree in electronics and information engineering from Xi'an Jiaotong University, Xi'an, in 2008. From 2010 to 2011, he was a Visiting Scholar with Microsoft Research Asia, Beijing, China. He was an Assistant Professor with Xi'an Jiaotong University, where he was an Associate Professor from 2011 to 2014, and is currently a Full Professor. He is also the Director of the the Smiles Laboratory, Xi'an Jiaotong University. His research interests include social media Big Data mining and search.