Contrastive Learning with Frequency-Domain Interest Trends for Sequential Recommendation

Yichi Zhang
zhangyichi@hrbeu.edu.cn
Harbin Engineering University
Harbin, Heilongjiang, China

Guisheng Yin*
yinguisheng@hrbeu.edu.cn
Harbin Engineering University
Harbin, Heilongjiang, China

Yuxin Dong*
dongyuxin@hrbeu.edu.cn
Harbin Engineering University
Harbin, Heilongjiang, China

ABSTRACT

Recently, contrastive learning for sequential recommendation has demonstrated its powerful ability to learn high-quality user representations. However, constructing augmented samples in the time domain poses challenges due to various reasons, such as fast-evolving trends, interest shifts, and system factors. Furthermore, the F-principle indicates that deep learning preferentially fits the low-frequency part, resulting in poor performance on high-frequency tasks. The complexity of time series and the low-frequency preference limit the utility of sequence encoders. To address these challenges, we need to construct augmented samples from the frequency domain, thus improving the ability to accommodate events of different frequency sizes. To this end, we propose a novel Contrastive Learning with Frequency-Domain Interest Trends for Sequential Recommendation (CFIT4SRec). We treat the embedding representations of historical interactions as "images" and introduce the second-order Fourier transform to construct augmented samples. The components of different frequency sizes reflect the interest trends between attributes and their surroundings in the hidden space. We introduce three data augmentation operations to accommodate events of different frequency sizes: low-pass augmentation, high-pass augmentation, and band-stop augmentation. Extensive experiments on four public benchmark datasets demonstrate the superiority of CFIT4SRec over the state-of-the-art baselines. The implementation code is available at https://github.com/zhangyichi1Z/CFIT4SRec.

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

KEYWORDS

Sequential Recommendation, Frequency domain, Contrastive Learning, Recommender System

ACM Reference Format:
Yichi Zhang, Guisheng Yin, and Yuxin Dong. 2023. Contrastive Learning with Frequency-Domain Interest Trends for Sequential Recommendation. In *Seventeenth ACM Conference on Recommender Systems (RecSys '23), September 18–22, 2023, Singapore, Singapore.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3604915.3608790

*Corresponding authors.

RecSys '23, September 18–22, 2023, Singapore, Singapore

ACM ISBN 979-8-4007-0241-9/23/09…$15.00
https://doi.org/10.1145/3604915.3608790

1 INTRODUCTION

Sequential recommendations aim to understand users' intention from historical interactions, and predict the next item that users are most likely to interact with in the future. Due to the rapid progress of deep learning, most existing sequential recommendations focus on learning to generate sequence representations via various deep network architectures, such as convolutional neural networks (CNNs) [17, 24], recurrent neural networks (RNNs) [8, 23], graph neural networks (GNNs) [13, 22] and Transformer [10, 16]. In particular, Transformer, which emphasizes the items most relevant to the target, shows its powerful domination as a sequence encoder for the representation model in sequential recommendation tasks. While these methods achieve promising results, the sparsity and noisy interactions in historical interactions limit the model's ability to understand users' intentions, leading to the poor inference of the next item.

To mitigate these issues, contrastive learning [12, 14, 19, 20, 25, 29] has been introduced to the sequential recommendation. Contrastive learning, a popular type of contrastive self-supervised learning, aims to learn a high-quality encoder by maximizing the consistency of the positive sample (i.e., view, signals) pairs and pushing the negative sample pairs apart. Thanks to the success of contrastive learning in various fields, CL4SRec [20] applies contrastive learning to construct self-supervised signals from raw sequences, thus improving encoder quality. DuoRec [14] proposes a supervised augmentation strategy where sequences with the same target item are regarded as positive examples. Such contrastive learning tasks can improve the discriminative ability of the encoder, thereby alleviating data sparsity and data noise.

However, unfortunately, the expected improvements are often not realized due to various reasons, such as fast-evolving trends, interest shifts, and system factors [26]. The complexity inherent in time series data can make it difficult to discern users' intentions accurately. Furthermore, the F-principle [21] indicates that deep learning preferentially fits the low-frequency part (smooth trend), resulting in poor performance on high-frequency tasks (transient events). To address these issues, we expect to construct self-supervised samples from a frequency domain perspective, as shown in Fig. 1a, thus improving the ability to discriminate between events of different frequency sizes. In addition, most previous approaches ignore the effect of the trend of hidden-dimensional features on the user interest pattern, as shown in Fig. 1b. The interest trend refers to the difference between a certain feature attribute in the hidden

space and its surrounding values, indicating the user's attribute inclination at a particular moment.

Motivated by the above limitations, inspired by the Caser [17] and Fourier transform [15], we treat the embedding representations of historical interactions as "images" and introduce the second-order Fourier transform from the field of computer vision to construct self-supervised signals. We refer to the results of the second-order Fourier transform as the spatiotemporal frequency domain representation. As shown in Fig. 2, we observe that the low-frequency components of the "image" represent the smooth trend. In contrast, the high-frequency components represent the areas of dramatic change, indicating the degree of change tendency of each "pixel" with respect to its surrounding "pixels". Based on this observation, the spatiotemporal frequency domain representation is a collection of different degrees of attribute variation trends. In light of this, we propose a novel data augmentation strategy with interest trends (i.e. events of different frequency sizes), which controls the frequency size of spatiotemporal frequency domain representation via mask filtering. Then, we construct self-supervised signals that reflect different trend characteristics. Furthermore, we consider three types of augmentation operations: low-pass augmentation, high-pass augmentation, and band-stop augmentation. The low-pass augmentation suppresses the frequency component features with dramatic changes and retains the features with smooth trends, while the high-pass augmentation does the opposite. Band-stop augmentation refers to randomly removing a certain degree of trend features and then treating the rest as a self-supervised signal. Our design has the advantage that we augment data from the perspective of attributes, which are more expressive than features composed of item-level indexes. In addition, from the perspective of the spatiotemporal frequency domain, we further directly consider the trend among user feature attributes, which is beneficial to understand the user's intention tendency.

Specifically, to realize the above idea, we propose a novel method called **C**ontrastive Learning with **F**requency-Domain **I**nterest **T**rends for **S**equential **Rec**ommendation (**CFIT4SRec**). Our approach applies three augmentation operations to tune the degree of interest trends to obtain high-quality self-supervised signals. We subsequently feed these signals into a contrastive loss function to maximize the agreement of positive view pairs and improve the inference ability. Finally, CFIT4SRec adopts a multi-task learning framework to jointly contrastive learning and sequential recommendation learning tasks to obtain a more accurate user representation model.

Our technical contributions can be summarized as follows:

(1) To the best of our knowledge, it is the first time that a spatiotemporal frequency domain approach has been applied to construct self-supervised signals based on interest trends for the sequential recommendation, which are simple and efficient.

(2) We consider the tendency features between adjacent attributes from the perspective of the spatiotemporal frequency domain for the first time. This allows us to gain a deeper understanding of the dynamic interests and intentions at each moment.

(3) We propose a multi-task learning solution that combines recommendation tasks and contrastive learning tasks to learn high-quality user representations.

(4) Extensive experimental results on four datasets show that our proposed method outperforms state-of-the-art models, demonstrating our solution's superiority.

2 RELATED WORKS

2.1 Sequential Recommendation

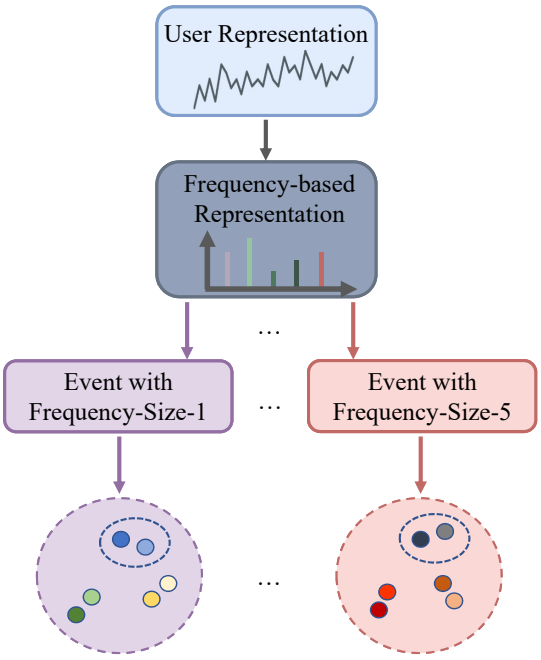
The sequential recommendation aims to explore users' interest patterns from their historical interactions. Early research [6] was mainly based on Markov chain assumptions and matrix factorization. With the proliferation of neural networks, deep learning techniques have been applied to the sequential recommendation. NCF [5] learns the interaction between users and items by replacing inner products with a multilayer perceptron. GRU4Rec [8] adopts the GRU network structure to learn long and short-term interest preferences from historical interactions. Caser [17] proposes a new perspective for processing historical interactions by treating the embedding representation as an "image", which captures sequential patterns from both a union and point-level perspective. SASRec [10] treats each interaction as an independent entity and then learns the pattern that significantly impacts the target behaviour with the self-attention mechanism. SR-GNN [22] processes the historical interactions from a graph perspective, treating the interactions as graph nodes and learning complex user preferences.

Unlike previous methods for sequential recommendation tasks, we analyze the historical interaction from the perspective of the spatiotemporal frequency domain, emphasizing the spatiotemporal variation trend features of attributes to understand the users' intentions.

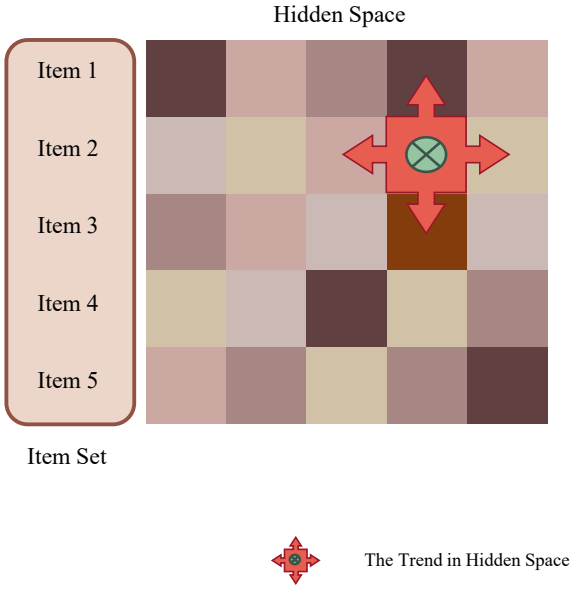
2.2 Contrastive Learning

Contrastive learning has recently achieved tremendous success in computer vision [1, 2] and natural language processing [3, 4]. Its core idea is to construct self-supervised samples for network training, thereby improving the inference ability of the model. Various works have recently adopted contrastive learning methods to make more suitable recommendations. S^3 Rec [29] adopts a random masking strategy to generate self-supervised samples at the attribute level, achieving good results. CL4SRec [20] borrows from natural language processing methods and applies three strategies (e.g. Crop, Reorder and Mask) to generate self-supervised samples at the sequence level. CoSeRec [12] utilizes supervised information by calculating the similarity of sequence instances to obtain high-quality self-supervised samples. DuoRec [14] designs a supervised generation strategy where historical behaviour sequences with the same target item have similar semantics, achieving state-of-the-art results.

Compared with previous data augmentation strategies in sequential recommendations with contrastive learning, we introduce spatiotemporal frequency domain techniques from computer vision for the first time. With such techniques, we design data augmentation strategies with interest trends to construct higher-quality self-supervised samples, which allow the proposed model to accommodate events of different frequency sizes.



(a) Contrastive learning for different frequency events.



(b) An illustration of the trend in the proposed model.

Figure 1: Motivation. (a) demonstrates that we expect to improve discrimination by constructing contrastive learning tasks with different frequency sizes. (b) explains our expectation to consider the trend of interest attributes in the hidden space with surrounding attributes in sequential recommendation.

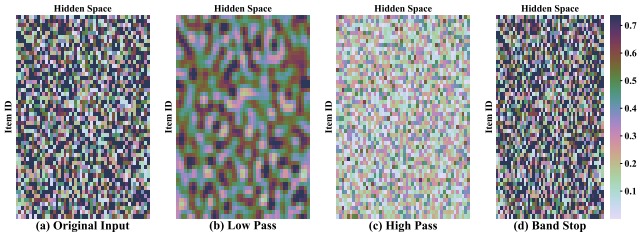


Figure 2: Visualization of three different frequency sizes, including low frequency, high frequency and band stop.

2.3 Frequency Domain

The frequency domain is a coordinate system that describes the characteristics of signals based on their frequency. Frequency domain analysis is a method that transforms a signal from the time domain to the frequency domain (e.g., Fourier transform algorithm), thus helping one to understand the characteristics from another perspective. Furthermore, spatiotemporal frequency domain analysis studies 2-dimensional signals. The frequency spectrum includes components of different frequency sizes and provides more intuitive and richer information than the time-domain signal.

However, the F-principle [21] implies that deep learning preferentially fits low-frequency components, resulting in poor performance in high-frequency tasks. In addition, AFMN [27] shows that different tasks focus on the different frequency components. Therefore, we expect sequential recommendations to accommodate events with frequencies of different size, thus improving discrimination in complex tasks. Taking this idea into account, we construct self-supervised samples for contrastive learning from the frequency domain perspective to learn high-quality encoders.

3 PRELIMINARIES

3.1 Notations and Task Definition

In the sequential recommendation, we adopt $\mathcal{U} = \{u_1, u_2, \cdots, u_{|\mathcal{U}|}\}$ to denote the set of users, and $\mathcal{V} = \{v_1, v_2, \cdots, v_{|\mathcal{V}|}\}$ to denote the pool of all unique items. For each user u, there exists a chronologically ordered historical interaction record $S^u = \left[v_1^u, v_2^u, \cdots, v_{|S^u|}^u\right]$, where $v_t^u \in \mathcal{V}$ represents the item that user u interacted with at time t, and $|S^u|$ denotes the number of interactions.

The task of sequential recommendation is to predict the next item $v_{|S^u|+1}^u$ that the user u likely to interact with, formally defined as follows:

$$\arg\max_{v_t^u \in \mathcal{V}} P\left(v_{|S^u|+1}^u = v_i | S^u\right), \tag{1}$$

where the formula computes the probability that each candidate item in the pool will be interacted with at the next time step, and then selects the one with the highest probability for a recommendation.

3.2 Fourier Transform

Fourier transform [15] is an essential technique in signal processing and image analysis, which can map the time domain representation to the frequency domain view. In this paper, we introduce the second-order discrete Fourier transform (2D-DFT), defined as follows:

$$F(m,z) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi\left(\frac{mx}{M}+\frac{zy}{N}\right)}, \tag{2}$$

where j is the imaginary unit, $f(x,y) \in \mathbb{R}^{M\times N}$ is the "image" in the time domain, and $F(m,z) \in \mathbb{R}^{M\times N}$ is the "image" in the frequency domain. Besides, (x,y) and (m,z) represent their coordinates in an "image." We denote 2D-DFT as DFT$(\cdot)$. Given $F(m,z)$, we can reconstruct the original "image" $f(x,y)$ via inverse 2D-DFT, IDFT$(\cdot)$, as follows:

$$f(x,y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{z=0}^{N-1} F(m,z) e^{j2\pi\left(\frac{mx}{M}+\frac{zy}{N}\right)}. \tag{3}$$

In this paper, we apply the fast Fourier transform (FFT) [7] to efficiently compute DFT, reducing time complexity from $O\left(n^2\right)$ to $O\left(nlog\ n\right)$.The inverse fast Fourier transform (IFFT) works similarly.

FFT can convert the input "image" to a representation in the time-frequency domain, where different frequency-domain sizes represent the degree of variation trend between the original pixel and its surrounding points. In this paper, we design three data augmentations, including low-pass augmentation, high-pass augmentation, and band-stop augmentation, to construct high-quality self-supervised signals by controlling the frequency-domain size.

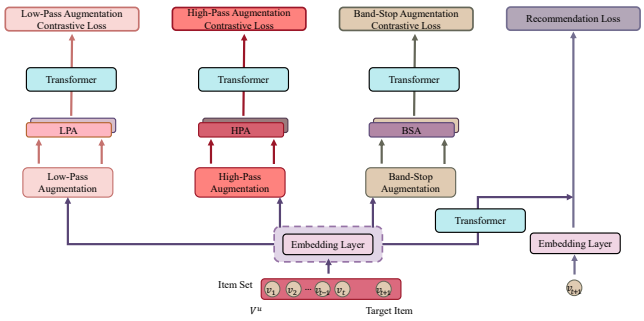


Figure 3: The overall architecture of the proposed CFIT4SRec model.

4 METHOD

In this section, we present the architecture of our proposed framework, Contrastive Learning with Frequency-Domain Interest Trends for Sequential Recommendation (CFIT4SRec). The architecture of CFIT4SRec is illustrated in Fig. 3.

4.1 Transformer as User Representation

Sequential recommendations aggregate users' historical interactions to characterize their preferences. Following state-of-the-art models such as DuoRec [14] and CL4SRec [20], we employ a Transformer to encode the sequence representation. The Transformer [18] is a powerful method for encoding sequences based on self-attention mechanisms and has achieved promising results in sequential recommendations. To efficiently encode the sequence, we first map the sequence representation to a hidden space with an embedding layer. Then, we model the relationships between items within the sequence via multi-head self-attention modules.

4.1.1 The Embedding Layer. The embedding layer maps the ID sequence of historical interactions to a d-dimensional hidden space with an embedding matrix $E \in \mathbb{R}^{(|\mathcal{V}|+1)\times d}$, thus efficiently encoding items. Furthermore, to preserve the temporal properties of the sequence, we introduce a learnable position embedding matrix $P \in \mathbb{R}^{T\times d}$, where T is the maximum length of the sequence. For the input original historical interaction sequence $S^u = \left[v_1^u, v_2^u, \cdots, v_T^u\right]$, we can obtain the embedding representation $I^u \in \mathbb{R}^{T\times d}$ with temporal features, formalized as follows:

$$I^u = \left[I_1^u, I_2^u, \cdots, I_T^u\right], \tag{4}$$

$$I_t^u = E_{v_t} + P_t, \tag{5}$$

where v_t^u is the item interacted by the user at time t and P_t is the location encoding at time t. In addition, we introduce Dropout and layer normalization to improve the robustness of the embedding representation and keep the training stable. The final output I^u is shown as follows:

$$I^u = \text{Dropout}(\text{LayerNorm}(I^u)). \tag{6}$$

4.1.2 Multi-Head Self-Attention Module. After obtaining the embedding representations, we apply the Transformer to capture the intrinsic relationships between their attributes. For example, for the input representation I^u, we stack L-layer multi-head self-attention in the Transformer encoder to generate a high-quality representation H^u, and the encoding procedure is shown below:

$$H^u = \text{Trm}^L(I^u), \tag{7}$$

where H^u aggregates the information of all items interacted before each time step t. Following previous work [14, 20], we choose the last hidden vector of h^u as the user's sequence representation, given by the following:

$$h^u = H^u[-1], \tag{8}$$

where -1 denotes the index of the last vector.

4.2 Recommendation Learning

The sequential recommendation can be treated as a classification task for the whole itemset, which predicts the likelihood of the next item. Given the user representation h and item embedding matrix E, the score $\hat{y} \in \mathbb{R}^{|\mathcal{V}|+1}$ of the next item is calculated as shown below:

$$\hat{y} = \text{softmax}(Eh), \tag{9}$$

where h is the pure user representation without FFT during inference. Of course, we can apply sampled softmax [9] if we need to scale to large dataset tasks. Furthermore, preliminary experiments

show that the sampled softmax speeds up the training but reduces the accuracy. Considering that the dataset involved in this paper is within an acceptable computational range and to make a fair comparison with other baselines, we employ the conventional softmax in this experiment.

Finally, we employ a binary cross-entropy loss to optimize the sequential recommendation task as follows:

$$\mathcal{L}_{\text{rec}} = -\sum_{v_i \in \mathcal{V}} (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)), \tag{10}$$

where y denotes the one-hot encoding vector of ground truth items.

4.3 Contrastive Learning

The complexity of time series and the low-frequency preference limit the utility of sequence encoders. To address these issues, we construct samples of different frequency sizes (i.e., different degrees of trend) for contrastive learning to improve the discriminative power of the encoder. Next, we present the proposed data augmentation operations and the contrastive loss.

4.3.1 Data Augmentation Operators. To realize our idea, we introduce three data augmentation operations: low-pass augmentation, high-pass augmentation, and band-stop augmentation.

First, we calculate the spatiotemporal frequency representation $Freq^u \in \mathbb{R}^{T \times d}$ of the user representation $I^u \in \mathbb{R}^{T \times d}$ as shown below:

$$Freq^u = \text{FShift}(\text{FFT}(I^u)), \tag{11}$$

where $\text{FFT}(\cdot)$ is the second-order fast Fourier transform, and $\text{FShift}(\cdot)$ moves the zero-frequency component of the frequency domain toward the center $(T/2, d/2)$. And then, the frequency size of $Freq^u$ increases from the center to the outer edge. In light of this, we can extract the desired frequency components, which is defined as follows:

$$Freq^u_{aug} = \text{Aug}(m, z) \cdot Freq^u, \tag{12}$$

where $\text{Aug}(\cdot)$ is the augmentation operation and (m, z) denotes the coordinate position in the frequency domain ($0 <= m <= T$ and $0 <= z <= d$). Next, we provide details of the three augmentation operations.

Low-Pass Augmentation (LPA). LPA adopts the low-frequency components as augmented samples, which represent the slowly evolving trend of the attributes and correspond to the central part of $Freq^u$. Thus, the LPA operation can be formalized as follows:

$$\text{LPA}(m, z) = \begin{cases} 1, D(m, z) \le D_0^L \\ 0, D(m, z) > D_0^L \end{cases}, \tag{13}$$

where D_0^L is the specified non-negative low-frequency threshold. $D(m, z)$ is the distance from point (m, z) to the center $(T/2, d/2)$ of the frequency domain, which is formalized as follows:

$$D(m, z) = \sqrt{(m - T/2)^2 + (z - d/2)^2}. \tag{14}$$

LPA suppresses drastically evolving trends in attributes and keeps the prime features of historical interactions. In addition, LPA can filter out the noise in the sequence.

High-Pass Augmentation (HPA). HPA treats the high-frequency component as augmented samples, representing the rapidly shifting trend in attributes and corresponding to the part of $Freq^u$ away from the center. Thus, the HPA strategy can be defined as follows:

$$\text{HPA}(m, z) = \begin{cases} 1, D(m, z) \ge D_0^H \\ 0, D(m, z) < D_0^H \end{cases}, \tag{15}$$

where D_0^H is the specified non-negative high-frequency threshold. HPA constructs samples of high-frequency components for contrastive learning, thus mitigating the low-frequency preference problem mentioned in the F-principle. Furthermore, it allows the sequence encoder to accommodate the rapidly evolving trend in attributes.

Band-Stop Augmentation (BSA). The BSA randomly suppresses a certain frequency component and takes the rest as augmented samples. Therefore, the BSA strategy is defined as follows:

$$\text{BSA}(m, z) = \begin{cases} 1, D(m, z) \ne D_0^B \\ 0, D(m, z) = D_0^B \end{cases}, \tag{16}$$

where D_0^B is a randomly generated non-negative value to suppress a certain frequency component. The BSA randomly masks a certain frequency component (i.e., the extent of the trend) while keeping the rest unchanged to mitigate overfitting.

Finally, we map the augmented frequency domain representation to the time domain and the augmented view in the time domain is formalized as follows:

$$TFreq^u_{aug} = \text{IFFT}(\text{Aug}(m, z) \cdot Freq^u),\ \text{Aug} \sim \{\text{LPA}, \text{HPA}, \text{BSA}\}, \tag{17}$$

where $\text{IFFT}(\cdot)$ is the inverse second-order Fourier fast transform.

4.3.2 Contrastive Loss. To improve encoder discrimination, contrastive learning maximizes the agreements between positive pairs from the same instance. To achieve this goal, we require a contrastive loss function that minimizes the difference between two augmented samples from the same sequence and maximizes the difference between two augmented samples from different sequences. Specifically, given a batch of sequences of size N, denoted as $[I^{u_1}, I^{u_2}, \ldots, I^{u_N}]$,we apply the augmentation operation with different Dropouts to each sequence, and finally obtain the augmented pairs sequence of size $2N$ as follows:

$$\left[I^{u_1}_{a1}, I^{u_1}_{a2}, I^{u_2}_{a1}, I^{u_2}_{a2}, \ldots, I^{u_N}_{a1}, I^{u_N}_{a2}\right],\ a \sim \{\text{LPA}, \text{HPA}, \text{BSA}\}, \tag{18}$$

where $(I^{u_1}_{a1}, I^{u_1}_{a2})$ represents a positive pair from the same sequence, and the other $2(N-1)$ samples are considered as negative pairs for this pair, denoted as S^{neg}. Then, we encode these samples via a shared Transformer based on Eq. (7) and Eq. (8) to obtain user representations as follows:

$$\left[h^{u_1}_{a1}, h^{u_1}_{a2}, h^{u_2}_{a1}, h^{u_2}_{a2}, \ldots, h^{u_N}_{a1}, h^{u_N}_{a2}\right],\ a \sim \{\text{LPA}, \text{HPA}, \text{BSA}\}. \tag{19}$$

For each positive pair $(h^{u_N}_{a1}, h^{u_N}_{a2})$, we apply the softmax cross entropy loss to optimize the encoder, formalized as follows:

$$\mathcal{L}_{cl}\left(h^{u_N}_{a1}, h^{u_N}_{a2}\right) = -\log \frac{\exp\left(\text{sim}\left(h^{u_N}_{a1}, h^{u_N}_{a2}\right)\right)/\tau}{\exp\left(\text{sim}\left(h^{u_N}_{a1}, h^{u_N}_{a2}\right)\right)/\tau + \sum_{s^- \in S^{\text{neg}}} \exp\left(\text{sim}\left(h^{u_N}_{a1}, s^-\right)\right)/\tau}, \tag{20}$$

where $\text{sim}(\cdot)$ is the dot product to measure the similarity between two augmented samples, and τ is the temperature coefficient. Consequently, for the data augmentation operations proposed in this

paper, we define their contrastive losses as follows:

$$\mathcal{L}_{\text{LPA}} = \mathcal{L}_{cl}(h^u_{a1}, h^u_{a2}),\ a \sim \text{LPA}, \tag{21}$$

$$\mathcal{L}_{\text{HPA}} = \mathcal{L}_{cl}(h^u_{a1}, h^u_{a2}),\ a \sim \text{HPA}, \tag{22}$$

$$\mathcal{L}_{\text{BSA}} = \mathcal{L}_{cl}(h^u_{a1}, h^u_{a2}),\ a \sim \text{BSA}\,. \tag{23}$$

Finally, we expect the user representation encoder to accommodate different frequency components (i.e., different degrees of interest trends) and thus improve the inference power. We simply linearly weight the above losses to obtain the final contrastive loss as follows:

$$\mathcal{L}_{\text{ssl}} = \mathcal{L}_{\text{LPA}} + \mathcal{L}_{\text{HPA}} + \mathcal{L}_{\text{BSA}}. \tag{24}$$

4.4 Multi-task Training

Both recommendation learning and contrastive learning optimize the user representation encoder, which models items within sequences; consequently, we adopt a multi-task learning framework to calculate the total contrastive loss in a linearly weighted way as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \lambda \mathcal{L}_{\text{ssl}} + \gamma \|\Theta\|_2^2\,, \tag{25}$$

where the hyperparameter λ controls the strength of contrastive learning, γ is the hyperparameter controlling the strength of the L_2 regularization, and Θ is the set of model parameters.

5 EXPERIMENTS

We conduct extensive experiments on four publicly available datasets to investigate the following:

1. How does the proposed CFIT4SRec perform compared to the state-of-the-art baselines?

2. How do different data augmentation operations impact the performance?

3. Does CFIT4SRec perform robustly against data sparsity and noisy interactions?

4. How do different hyperparameter settings affect the performance of CFIT4SRec?

5.1 Experimental Setup

5.1.1 Datasets. We evaluate the performance of CFIT4SRec on four publicly available datasets from real-world platforms. The ***Amazon***[1] dataset is collected from the user interaction records of the Amazon e-commerce platform, which is widely used to evaluate various sequential recommendation algorithms. We employ three of its sub-datasets, namely ***Beauty***, ***Clothing*** and ***Sports***, to evaluate the proposed approach. ***MovieLens***[2] is a movie rating dataset that contains real user interaction information. We adopt the ***MovieLens-1M*** version in our experiments.

Following previous work [12, 14, 19, 20], we ignore duplicate interactions and filter out users and items with less than 5 recorded interactions. Then we sort the historical interactions chronologically. Similar to recent work [14, 20], we employ a leave-one-out strategy to partition the datasets, where the last item in the sequence is employed as the test set, the second last item is treated as the validation set, and the rest is used for training. The statistics of the processed datasets are summarized in Table 1.

[1]http://jmcauley.ucsd.edu/data/amazon/

[2]https://grouplens.org/datasets/movielens

Table 1: Statistics of the datasets.

Dataset	#users	#items	#actions	avg.actions/user	avg.actions/item	Sparsity
Beauty	22364	12102	198502	8.9	16.4	99.93%
Clothing	39388	23034	278677	7.1	12.1	99.97%
Sports	35599	18358	296337	8.3	16.1	99.95%
ML-1M	6041	3417	999611	165.5	292.6	95.16%

5.1.2 Evaluation Metrics. To evaluate the performance of all methods, we adopt the popular ranking metrics, including hit rate (HR) and normalized discounted cumulative gain (NDCG). HR@K focuses on the presence of positive samples, while NDCG@K further considers position ranking information. We evaluate the ranking results over the whole item set for a fair comparison. In this work, we report HR and NDCG with K = {5,10}.

5.1.3 Competing Models. To demonstrate the effectiveness of our proposed model, we first consider the following non-contrastive learning sequential recommendation models:

GRU4Rec [8] applies GRU as an encoder to model user representations with long and short-term features.

Caser [17] adopts horizontal and vertical convolutions to capture high-order interest patterns for the sequential recommendation.

SASRec [10] adopts a unidirectional multi-head self-attention mechanism to learn high-quality user representations.

BERT4Rec [16] introduces a bidirectional Transformer as an encoder and employs the cloze task to train the model.

We then consider the recent contrastive sequential recommendation models:

$\mathbf{S^3Rec_{MIP}}$ [29] is a sequential recommendation with contrastive learning based on maximizing mutual information. In this work, we adopt the Mask Item Prediction (MIP) variant.

CL4SRec [20] combines contrastive learning and sequential recommendation models to learn high-quality sequence encoder.

DuoRec [14] proposes a supervised data augmentation strategy, where sequences with the same target items have similar semantics to alleviate representation degradation problems.

5.1.4 Implementation Details. For each baseline, we adopt the implementation provided by the authors. To keep a fair comparison, we set the embedding size to 64 and the batch size to 256. We follow the instructions from the original paper to set other hyperparameters. We implement our model in the popular RecBole [28] recommendation system framework, PyTorch 1.8.2, and Python 3.7.10. We adopt the Adam [11] optimizer with a learning rate of 0.001. We employ an early stopping strategy, where if HR@10 does not improve within 10 training epochs, the training is stopped. The parameter γ that control the L_2 regularization are searched in the range {0, 0.001, 0.0001}. The number of layers and heads in the Transformer is set to 2. We tune the Dropout parameter within {0.1, 0.3, 0.5, 0.7, 0.9}. The parameter λ in Eq. (25) is selected from the set {0.05, 0.1, 0.3, 0.5, 0.7, 0.9}. The thresholds for LPA and HPA are chosen from {6, 12, 18, 24}. The range of parameter τ is {0.1, 1, 6}. All models are trained on a single NVIDIA GeForce RTX 2060S GPU.

5.2 Overall Performance Comparison (RQ1)

Table 2 presents the overall performance of all methods on four public datasets, where the best performance is shown in bold and

Table 2: Overall performance of different methods. All improvements are statistically significant.

Dataset	Metric	GRU4Rec	Caser	SASRec	S^3 Rec_{MIP}	CL4SRec	DuoRec	CFIT4SRec	Improv.
Beauty	HR@5	0.0159	0.0241	0.0380	0.0334	0.0396	0.0516	**0.0564**	9.30%
	HR@10	0.0367	0.0413	0.0625	0.0609	0.0652	0.0791	**0.0831**	5.06%
	NDCG@5	0.0082	0.0120	0.0237	0.0193	0.0229	0.0335	**0.0347**	3.58%
	NDCG@10	0.0154	0.0239	0.0301	0.0281	0.0332	0.0413	**0.0434**	5.08%
Clothing	HR@5	0.0091	0.0108	0.0137	0.0119	0.0131	0.0181	**0.0195**	7.73%
	HR@10	0.0145	0.0156	0.0234	0.0205	0.0225	0.0279	**0.0293**	5.02%
	NDCG@5	0.0058	0.0067	0.0075	0.0068	0.0080	0.0105	**0.0111**	5.71%
	NDCG@10	0.0076	0.0090	0.0102	0.0098	0.0113	0.0137	**0.0143**	4.38%
Sports	HR@5	0.0150	0.0156	0.0213	0.0185	0.0235	0.0302	**0.0330**	9.27%
	HR@10	0.0261	0.0294	0.0313	0.0324	0.0379	0.0455	**0.0497**	9.23%
	NDCG@5	0.0086	0.0096	0.0115	0.0117	0.0137	0.0179	**0.0185**	3.35%
	NDCG@10	0.0137	0.0147	0.0168	0.0158	0.0181	0.0227	**0.0239**	5.29%
ML-1M	HR@5	0.0813	0.0871	0.1049	0.1061	0.1221	0.1729	**0.2157**	24.75%
	HR@10	0.1598	0.1743	0.1817	0.1812	0.1965	0.2746	**0.3096**	12.75%
	NDCG@5	0.0489	0.0535	0.0672	0.0639	0.0746	0.1249	**0.1479**	18.41%
	NDCG@10	0.0711	0.0819	0.0886	0.0847	0.0968	0.1493	**0.1771**	18.62%

Table 3: Analysis on different data augmentation operations.

Augmentation			Beauty		Clothing		Sports		ML-1M	
LPA	HPA	BSA	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5
0	0	0	0.0380	0.0237	0.0137	0.0075	0.0213	0.0115	0.1049	0.0672
0	0	1	0.0527	0.0326	0.0184	0.0102	0.0310	0.0174	0.2083	0.1400
0	1	0	0.0541	0.0321	0.0190	0.0103	0.0322	0.0180	0.2084	0.1420
1	0	0	0.0549	0.0323	0.0179	0.0097	0.0320	0.0175	0.2103	0.1444
0	1	1	0.0558	0.0333	0.0185	0.0104	0.0327	0.0182	0.2126	0.1433
1	0	1	0.0559	0.0336	0.0183	0.0102	0.0319	0.0180	0.2151	0.1467
1	1	0	0.0536	0.0323	0.0186	0.0102	0.0318	0.0179	0.2127	0.1441
1	1	1	**0.0564**	**0.0347**	**0.0195**	**0.0111**	**0.0330**	**0.0185**	**0.2157**	**0.1479**

the second best is underlined. We can make some interesting observations:

Sequential recommendations with self-attention (e.g., SASRec, BERT4Rec, S^3 Rec_{MIP}, CL4SRec, DuoRec, and CFIT4SRec) consistently outperform GRU4Rec and Caser on all datasets. We attribute such improvement to the self-attention mechanism, which focuses on items that really influence the target behaviour.

Compared with conventional methods, sequential recommendations with contrastive learning improve the performance by a significant margin in most cases on all datasets, which demonstrates that contrastive learning can learn a high-quality sequence encoder to provide more accurate recommendations. In particular, the performance of S^3 Rec_{MIP} is slightly lower than SASRec. We suspect that it is caused by a two-stage training process, where information is not shared between contrastive learning and recommendation learning.

CFIT4SRec shows significant improvements over the state-of-the-art models on all datasets. In particular, our model achieves a 12.75% to 24.75% improvement in HR and NDCG compared to the strongest baseline on ML-1M. Such results demonstrate the superiority of our solution. We attribute such improvement to the proposed data augmentation operations (low-pass augmentation, high-pass augmentation, and band-stop augmentation), which make the user representation encoder accommodate different frequency components (i.e., different degrees of interest trends) and thus improve the inference power.

5.3 Impact of different data augmentation operations (RQ2)

To study the impact of different data augmentation operations, we design multiple variants for ablation experiments. For all models, we keep the optimal parameters unchanged on each dataset and only tune the combination of augmentation operations. We report the HR@5 and NDCG@5 results for the six variants and two default models on four datasets in Table 3, where the best results are bolded. The two default models are (LAP, HPA, BSA) = (0, 0, 0), indicating the SASRec without the contrastive learning framework, and (LAP, HPA, BSA) = (1, 1, 1), indicating the proposed CFIT4SRec with the three augmentation operations. Based on the results of the ablation experiments, we can draw a few interesting observations.

From Table 3, we observe that the model with contrastive learning consistently outperforms SASRec on all datasets, which confirms the effectiveness of contrastive learning. Furthermore, we also observe a better average performance for the model with more augmentation operations. We attribute such improvement to well-designed augmentation operations, which comprehensively accommodate the user representation's low-frequency, high-frequency, and band-stop cases. Besides, we note that different datasets focus on different frequency components. For example, Beauty and ML-1M concentrate more on the low-frequency component, so the model with the low-frequency augmentation operation outperforms the other models. In contrast, Clothing and Sports prefer the high-frequency component. We suspect that the frequency components emphasized by the model depend on the dataset's configuration, such as sparsity and the speed of interest evolution. We will further investigate this effect in future work. Finally, we observe that the additional BSA can further improve the performance in most cases, which demonstrates the effectiveness of the BSA.

5.4 Robustness Analysis (RQ3)

We conduct extensive robustness experiments to evaluate the proposed model against data sparsity and noisy interactions.

Data sparsity analysis experiment. To simulate data sparsity scenarios, we keep the test dataset unchanged and only tune the percentage of the training dataset (25%, 50%, 75%, and 100%). We compare the performance of the proposed model with the strongest baseline DuoRec. Fig. 4 reports the scores of NDCG@10 on ML-1M and Beauty. We observe that the proposed model consistently outperforms DuoRec on all datasets. In addition, the performance degradation of CFIT4SRec is slower than DuoRec. For example, when the training dataset is only 50% on ML-1M, DuoRec drops 25.59% of its original performance while the proposed model only drops 13.33%. Such results demonstrate the superiority of our solution. We suspect that the sparsity of the original dataset affects performance improvement against data sparsity. For example, with a 50% training dataset on Beauty, the DuoRec drops 48.91% while CFIT4SRec only drops 44.70%.

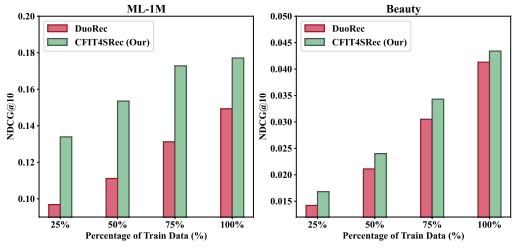


Figure 4: Performance comparison w.r.t. train ratio in HDCG@10.

Noise interactions analysis experiment. To evaluate the effect of noise interactions on the model, we train the model on the original training dataset and randomly replace a certain percentage (5%, 10%, 15%, and 20%) of items in each test sequence with negative user-item interactions. Significantly, we keep the maximum sequence length the same. Fig. 5 shows the performance of the proposed model and DuoRec on ML-1M and Beauty. We observe that the proposed model consistently outperforms DuoRec on all datasets, which demonstrates the superiority of our solution again. In addition, as the percentage of noise interactions on ML-1M and Beauty increased, the average degradation rates were 10.27% and 16.86% for CFIT4SRec respectively, while 16.32% and 26.54% for DuoRec. We attribute such improvement to the proposed augmentation method, which provide more confident positive samples for contrastive learning to improve the inference ability.

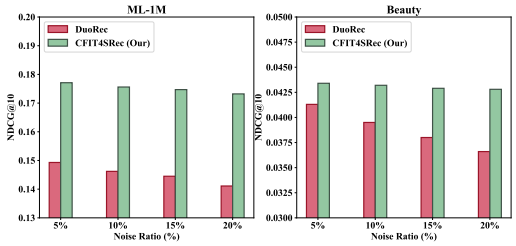


Figure 5: Performance comparison w.r.t. noise ratio in HDCG@10.

5.5 Hyperparameter Sensitivity (RQ4)

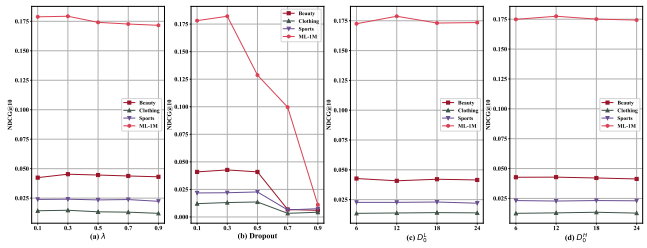


Figure 6: Performance comparison w.r.t different hyperparameters in HDCG@10.

We study the influence of three important hyperparameters in CFIT4SRec, including the strength of contrastive learning λ, the Dropout for positive pairs, the threshold D_0^L for LPA, and the threshold D_0^H for HPA . Fig. 6 reports the results on all datasets.

The strength of contrastive learning. A larger λ indicates a larger proportion of contrastive learning in the sequential recommendation. From Fig. 6a, we observe that the performance drops sharply once λ increases to a certain threshold, which means that it will degrade the efficiency of sequential recommendations when contrastive learning dominates the learning process. Such a result suggests that an appropriate strength of contrastive learning can improve the inference ability of the model.

The Dropout for positive pairs. We investigate the Dropout effect for positive pairs since it can change the frequency during the training process. Fig. 6b shows that a smaller appropriate Dropout can improve the performance, which does not intrinsically modify the frequency. In contrast, a Dropout over a certain threshold range (0.3 or 0.5 in our experiments) leads to a sharp degradation in

performance. We suspect that a larger Dropout causes an intrinsic change to the frequency of the original sequence. We will analyze this impact carefully in future work.

The thresholds for LPA and HPA. The thresholds control the division boundaries of the frequency components. Fig. 6c and Fig. 6d show the performance at different thresholds. We observe that either too large or too small thresholds limit the quality of the desired frequency domain components. In addition, the threshold range varies for different datasets. Such results suggest that the bounds of low and high frequencies need to be fine-tuned to achieve a good performance.

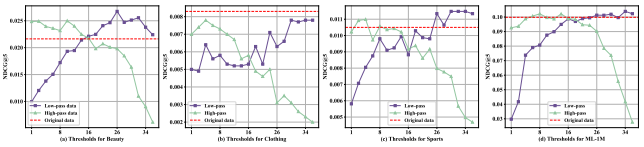


Figure 7: Performance comparison of Transformer w.r.t different filters in HDCG@5.

6 FURTHER ANALYSIS

Our proposed CFIT4SRec contains different filters for data augmentation during the training process. In addition, the different second-order frequency events represent the trends of interest mentioned above. To verify the effects of filters on the input for the transformer layers, we conduct the ablation study without contrastive learning when training a recommender on low- or high-pass data only. Fig. 7 reports the NDCG@5 results on four datasets.

From Fig. 7, we observe that more high or low frequencies achieve better performance, which means that we need to make the encoder adapt to different frequencies. In addition, we observe that the filtered data with a suitable threshold realize a performance close to or even exceeding the original data in most cases. Such results indicate that the filtered data can be utilized as self-supervised samples for training. Furthermore, when a certain threshold is exceeded, more high or low frequencies lead to a degradation of the model performance. We speculate that noise exists in both high and low frequencies. In conclusion, the sequence encoder should accommodate different frequency components (i.e. different interest trends) and thus improve its inference ability.

7 CONCLUSION AND FUTURE WORK

In this work, we propose a novel Contrastive Learning with Frequency-Domain Interest Trends for Sequential Recommendation (CFIT4SRec), which mitigates the complexity of time series and low-frequency preference. We introduce second-order frequency domain representations to construct positive samples for contrastive learning, which allows the sequence encoder to accommodate different frequency components and improve its inference ability. The components of different frequency sizes reflect the interest trends between attributes and their surroundings in the hidden space. Thus, we can strengthen the ability to discriminate features with different degrees of interest trends. Extensive experiment results on the four public benchmark datasets show that our CFIT4SRec achieves significant improvements and outperforms the start-of-the-art baselines.

Limitations and future directions. Simple frequency domain augmentation operations limit the flexibility to accommodate complex tasks. We need to explore more advanced frequency-domain augmentation methods. Several directions remain to be explored. How do we design learnable augmentation methods from the frequency domain perspective? Moreover, we can further study how to add supervised information to construct high-quality augmented samples.

REFERENCES

[1] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.). https://proceedings.neurips.cc/paper/2020/hash\/70feb62b69f16e0238f741fab228fec2-Abstract.html

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 1597–1607. http://proceedings.mlr.press/v119/chen20j.html

[3] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, 6894–6910. https://doi.org/10.18653/v1/2021.emnlp-main.552

[4] Beliz Gunel, Jingfei Du, Alexis Conneau, and Veselin Stoyanov. 2021. Supervised Contrastive Learning for Pre-trained Language Model Fine-tuning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net. https://openreview.net/forum?id=cu7IUiOhujH

[5] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, Rick Barrett, Rick Cummings, Eugene Agichtein, and Evgeniy Gabrilovich (Eds.). ACM, 173–182. https://doi.org/10.1145/3038912.3052569

[6] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. 2017. Fast Matrix Factorization for Online Recommendation with Implicit Feedback. *CoRR* abs/1708.05024 (2017). arXiv:1708.05024 http://arxiv.org/abs/1708.05024

[7] M. T. Heideman, D. H. Johnson, and C. S. Burrus. 1985. Gauss and the history of the fast Fourier transform. *Archive for History of Exact ences* 34, 3 (1985), 265–277.

[8] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1511.06939

[9] Sbastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On Using Very Large Target Vocabulary for Neural Machine Translation. In *PROCEEDINGS OF THE 53RD ANNUAL MEETING OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS AND THE 7TH INTERNATIONAL JOINT CONFERENCE ON NATURAL LANGUAGE PROCESSING, VOL 1*, C Zong and M Strube (Eds.). Assoc Computat Linguist; Asian Federat Nat Language Proc; CreditEase; Baidu; Tencent; Alibaba Grp; Samsung; Microsoft; Google; Facebook; SinoVoice; Huawei; Nuance; Amazon; Voicebox Technologies; Baobab; Sogou, 1–10. 53rd Annual Meeting of the Association-for-Computational-Linguistics (ACS) / 7th International Joint Conference on Natural Language Processing of the Asian-Federation-of-Natural-Language-Processing (IJCNLP), Beijing, PEOPLES R CHINA, JUL 26-31, 2015.

[10] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *IEEE International Conference on Data Mining, ICDM 2018, Singapore, November 17-20, 2018*. IEEE Computer Society, 197–206. https://doi.org/10.1109/ICDM.2018.00035

[11] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6980

[12] Zhiwei Liu, Yongjun Chen, Jia Li, Philip S. Yu, Julian J. McAuley, and Caiming Xiong. 2021. Contrastive Self-supervised Sequential Recommendation with Robust Augmentation. *CoRR* abs/2108.06479 (2021). arXiv:2108.06479 https://arxiv.org/abs/2108.06479

[13] Zhiqiang Pan, Fei Cai, Wanyu Chen, Honghui Chen, and Maarten de Rijke. 2020. Star Graph Neural Networks for Session-based Recommendation. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux (Eds.). ACM, 1195–1204. https://doi.org/10.1145/3340531.3412014

[14] Ruihong Qiu, Zi Huang, Hongzhi Yin, and Zijian Wang. 2022. Contrastive Learning for Representation Degeneration Problem in Sequential Recommendation. In *WSDM '22: The Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event / Tempe, AZ, USA, February 21 - 25, 2022*, K. Selcuk Candan, Huan Liu, Leman Akoglu, Xin Luna Dong, and Jiliang Tang (Eds.). ACM, 813–823. https://doi.org/10.1145/3488560.3498433

[15] S. S. Soliman and MD Srinath. 1990. Continuous and discrete signals and systems. *Prentice Hall,* (1990).

[16] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, Wenwu Zhu, Dacheng Tao, Xueqi Cheng, Peng Cui, Elke A. Rundensteiner, David Carmel, Qi He, and Jeffrey Xu Yu (Eds.). ACM, 1441–1450. https://doi.org/10.1145/3357384.3357895

[17] Jiaxi Tang and Ke Wang. 2018. Personalized Top-N Sequential Recommendation via Convolutional Sequence Embedding. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, Yi Chang, Chengxiang Zhai, Yan Liu, and Yoelle Maarek (Eds.). ACM, 565–573. https://doi.org/10.1145/3159652.3159656

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 5998–6008. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[19] Yinwei Wei, Xiang Wang, Qi Li, Liqiang Nie, Yan Li, Xuanping Li, and Tat-Seng Chua. 2021. Contrastive Learning for Cold-Start Recommendation. In *MM '21: ACM Multimedia Conference, Virtual Event, China, October 20 - 24, 2021*, Heng Tao Shen, Yueting Zhuang, John R. Smith, Yang Yang, Pablo Cesar, Florian Metze, and Balakrishnan Prabhakaran (Eds.). ACM, 5382–5390. https://doi.org/10.1145/3474085.3475665

[20] Xu Xie, Fei Sun, Zhaoyang Liu, Shiwen Wu, Jinyang Gao, Bolin Ding, and Bin Cui. 2020. Contrastive Learning for Sequential Recommendation. https://doi.org/10.48550/ARXIV.2010.14395

[21] Zqj Xu, Y. Zhang, T. Luo, Y. Xiao, and Z. Ma. 2020. Frequency Principle: Fourier Analysis Sheds Light on Deep Neural Networks. *Communications in Computational Physics* 5 (2020).

[22] Gang Yang, Xiaofeng Zhang, and Yueping Li. 2020. Session-Based Recommendation with Graph Neural Networks for Repeat Consumption. In *ICCPR 2020: 9th International Conference on Computing and Pattern Recognition, Xiamen, China, October 30 - Vovember 1, 2020*. ACM, 519–524. https://doi.org/10.1145/3436369.3436454

[23] Haochao Ying, Fuzhen Zhuang, Fuzheng Zhang, Yanchi Liu, Guandong Xu, Xing Xie, Hui Xiong, and Jian Wu. 2018. Sequential Recommender System based on Hierarchical Attention Networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, Jérôme Lang (Ed.). ijcai.org, 3926–3932. https://doi.org/10.24963/ijcai.2018/546

[24] Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M. Jose, and Xiangnan He. 2019. A Simple Convolutional Generative Network for Next Item Recommendation. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, February 11-15, 2019*, J. Shane Culpepper, Alistair Moffat, Paul N. Bennett, and Kristina Lerman (Eds.). ACM, 582–590. https://doi.org/10.1145/3289600.3290975

[25] Xu Yuan, Hongshen Chen, Yonghao Song, Xiaofang Zhao, and Zhuoye Ding. 2021. Improving Sequential Recommendation Consistency with Self-Supervised Imitation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, Zhi-Hua Zhou (Ed.). ijcai.org, 3321–3327. https://doi.org/10.24963/ijcai.2021/457

[26] Xiang Zhang, Ziyuan Zhao, Theodoros Tsiligkaridis, and Marinka Zitnik. 2022. Self-Supervised Contrastive Pre-Training For Time Series via Time-Frequency Consistency. In *Advances in Neural Information Processing Systems*, Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (Eds.). https://openreview.net/forum?id=OJ4mMfGKLN

[27] Yichi Zhang, Guisheng Yin, Hongbin Dong, and Liguo Zhang. 2022. Attention-based Frequency-aware Multi-scale Network for Sequential Recommendation. *Applied Soft Computing* 127 (2022), 109349. https://doi.org/10.1016/j.asoc.2022.109349

[28] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. 2021. RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, Gianluca Demartini, Guido Zuccon, J. Shane Culpepper, Zi Huang, and Hanghang Tong (Eds.). ACM, 4653–4664. https://doi.org/10.1145/3459637.3482016

[29] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-Rec: Self-Supervised Learning for Sequential Recommendation with Mutual Information Maximization. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux (Eds.). ACM, 1893–1902. https://doi.org/10.1145/3340531.3411954