Modeling User Fatigue for Sequential Recommendation

Nian Li*
Shenzhen International Graduate School, Tsinghua University
Shenzhen, China

Xin Ban*
Cheng Ling
Kuaishou Inc.
Beijing, China

Chen Gao†
Tsinghua University
Beijing, China

Lantao Hu
Peng Jiang
Kuaishou Inc.
Beijing, China

Kun Gai
Independent
Beijing, China

Yong Li†
Tsinghua University
Beijing, China

Qingmin Liao
Shenzhen International Graduate School, Tsinghua University
Shenzhen, China

ABSTRACT

Recommender systems filter out information that meets user interests. However, users may be tired of the recommendations that are too similar to the content they have been exposed to in a short historical period, which is the so-called *user fatigue*. Despite the significance for a better user experience, user fatigue is seldom explored by existing recommenders. In fact, there are three main challenges to be addressed for modeling user fatigue, including what features support it, how it influences user interests, and how its explicit signals are obtained. In this paper, we propose to model user **F**atigue in interest learning for sequential **Rec**ommendations (**FRec**). To address the first challenge, based on a multi-interest framework, we connect the target item with historical items and construct an interest-aware similarity matrix as features to support fatigue modeling. Regarding the second challenge, built upon feature cross, we propose a fatigue-enhanced multi-interest fusion to capture long-term interest. In addition, we develop a fatigue-gated recurrent unit for short-term interest learning, with temporal fatigue representations as important inputs for constructing update and reset gates. For the last challenge, we propose a novel sequence augmentation to obtain explicit fatigue signals for contrastive learning. We conduct extensive experiments on real-world datasets, including two public datasets and one large-scale industrial dataset. Experimental results show that FRec can improve AUC and GAUC up to 0.026 and 0.019 compared with state-of-the-art models, respectively. Moreover, large-scale online experiments demonstrate the effectiveness of FRec for fatigue reduction. Our codes are released at https://github.com/tsinghua-fib-lab/SIGIR24-FRec.

*Contribute equally to this work.
†Corresponding author (chgao96@gmail.com, liyong07@tsinghua.edu.cn).

SIGIR '24, July 14–18, 2024, Washington, DC, USA

ACM ISBN 979-8-4007-0431-4/24/07.
https://doi.org/10.1145/3626772.3657802

CCS CONCEPTS

• **Information systems** → **Information systems applications**.

KEYWORDS

User Fatigue; Sequential Recommendation; Long and Short-term Interests

ACM Reference Format:
Nian Li, Xin Ban, Cheng Ling, Chen Gao, Lantao Hu, Peng Jiang, Kun Gai, Yong Li, and Qingmin Liao. 2024. Modeling User Fatigue for Sequential Recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24), July 14–18, 2024, Washington, DC, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3626772.3657802

1 INTRODUCTION

In today's online platforms, the recommender system is broadly deployed to filter out irrelevant content and fetch personalized content for users that they are interested in [10, 11, 13, 15, 33]. Therefore, in the development of recommendation models, how to capture user interests as accurately as possible is an essential problem.

Sequential recommender organizes users' historical interactions in a temporal sequence and aims to predict the next item of interaction [6, 43]. Many existing works built upon advanced neural networks focus on interest learning, including long and short-term user interests [16, 35, 42, 43]. Some works also propose to combine long and short-term interest modeling for better recommendation [3, 42, 43]. Another line of modeling accurate user interests is to extract multiple interests from the sequence [5, 26, 31]. These works argue that only one representation for modeling interests is not effective enough since users are usually interested in several kinds of items.

Despite of this, user fatigue has not been well studied in existing works, especially how it can influence user interests. In this work, **user fatigue refers that users may be tired of the recommendations that are too similar to content they have been**

exposed to in a short historical period, such as news, advertisements, *etc.* For example, the click-through rate (CTR) of news will drop significantly with more and more times of exposure [38]. It is important to note that user fatigue is fundamentally different from other concepts related to positive user experience in recommender systems. Specifically, *diversity* typically focuses solely on the dissimilarity between items in the recommendation list, irrespective of the user's historical interactions [2]. On the other hand, *serendipity* or *novelty* emphasizes that the recommended items are unexpected or unknown to the user, characterized by their divergence from historical items or by the items' popularity [12]. In contrast, user fatigue represents the negative aspect of user experience. We verify the existence of user fatigue on a micro-video platform Kuaishou, using large-scale interaction data involving tens of millions of users. An industrial dataset is also collected from this platform for experiments in Section 4. In Figure 1, we plot the *normalized* effective view-through rate (EVTR) as the function of the number of effective views of videos with the same category in historical consumption. Compared with videos with other categories, the EVTR of videos with the target category decreases significantly and is consistently lower when users have too many effective views of the same category. This is obvious evidence of user fatigue with respect to the repetitive consumption of similar videos. This issue can harm user experience and further reduce platform activity.

A few existing works address the issue of user fatigue with coarse-grained features based on item-level and category-level repetitions. Ma *et al.* [29] just feed these features into decision trees, which serve as the base recommendation model. Moriwaki *et al.* [30] define a simple quadratic function for directly mapping the features to user fatigue. These methods are usually ineffective since the way of modeling fatigue lacks flexibility and interpretability. As a matter of fact, there are three challenges to be addressed,

- **Fine-grained features are hard to obtain to support fatigue modeling.** Intuitively, user fatigue depends on the similarity between the target and historical items. Existing works usually utilize item-level and category-level repetitions as the similarity features [29, 38]. However, these measurements are usually too coarse to represent the similarity between items accurately. For instance, even if the two videos both belong to the category of 'pandas', there may be still non-negligible differences, such as one is about 'panda is eating bamboo' and the other is about 'pandas rentals from the UK'. Therefore, how to measure fine-grained similarity to support fatigue modeling is critical but difficult.
- **The influence of user fatigue on interests is complex.** In general, the user's certain interest will be weakened if he/she is experiencing fatigue with it. Existing works either neglect to model this influence [29, 38] or manually define it by a quadratic function [30], which is unrealistic in real-world scenarios. Actually, multiple historical items may contribute to causing user fatigue as a whole and further influence both long and short-term interests. Therefore, based on similarity features, how to fuse user fatigue with interest learning is also an essential point.
- **There are no explicit signals of user fatigue contained in historical consumption.** The decreasing engagement with certain types of items over time can be seen as users are tired of frequent exposures. However, this phenomenon can only be observed from later consumption after the current interaction. Therefore, it is hard to directly obtain corresponding signals of user fatigue with respect to the current item from historical consumption.

In this work, we propose to model user fatigue in interest learning for sequential recommendations with the challenges above addressed. Specifically, we first extract multi-interest representations[1] from the historical sequence with the self-attention mechanism. To obtain fine-grained features to support fatigue modeling, we construct an interest-aware similarity matrix (ISM) measured by the projection distance built upon historical and target item embeddings. We then apply cross networks for feature interplay to assist in handling complex fatigue influence, based on which we model the influence on long-term interest. We further develop a fatigue-gated recurrent unit (FRU) for short-term interest learning. For explicit signals of user fatigue, we propose a novel sequence augmentation to obtain them counterfactually and use them to supervise contrastive learning with respect to fatigue prediction.

We have conducted extensive experiments on two public datasets and one large-scale industrial dataset to evaluate the effectiveness of our FRec. Compared with many state-of-the-art (SOTA) models, our FRec achieves significant improvements with respect to various accuracy and ranking metrics. Further online studies also demonstrate that FRec can reduce user fatigue by alleviating repeated exposure in consecutive consumptions and improve user experience significantly.

The contributions of this work are summarized as follows,

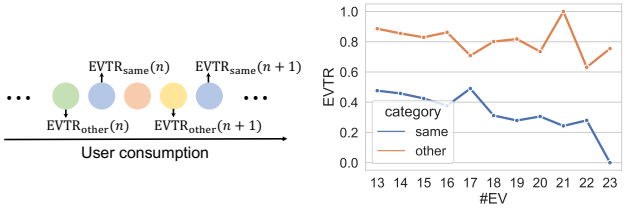
- We take user fatigue into consideration and make an advanced step to incorporate it into interest learning for sequential recommendations.
- We address primary challenges in modeling user fatigue by constructing fine-grained similarity features, handling its complex influence on long and short-term interests, and obtaining its signals with a novel sequence augmentation for contrastive learning.
- We conduct extensive offline and online experiments to demonstrate that FRec can improve the recommendation accuracy significantly (AUC up to 0.026, GAUC up to 0.019, and NDCG up to 5.8%) compared with SOTA methods and reduce user fatigue.

2 PROBLEM FORMULATION

We consider a standard problem of sequential recommendation. For each user $u \in \mathcal{U}$, let $S_u = \{i_1, i_2, \cdots, i_{L_u}\}$ denote the historical interaction sequence chronologically, *i.e.*, ordered by the interaction timestamp of each item, where $i_l \in \mathcal{I}$ and L_u is the sequence length. $\mathcal{U}$ and $\mathcal{I}$ denote the set of users and items, respectively.

Most existing works focus on modeling user interests according to historical sequence S_u for predicting whether the user will interact with the target item i_t. However, user fatigue is also a critical factor influencing the user interest and decision. In other words, if the item i_t is very similar to many items in S_u, the user may not interact with it due to being tired of repeated interactions. In this work, we aim to model user fatigue with the sequence S_u and incorporate it with interest learning for capturing user decisions more accurately.

[1]We use "representation" and "embedding" interchangeably in this paper.



(a) Illustration demo (b) EVTR trend

Figure 1: (a) An illustration demo to show how we measure EVTR along with user consumption. Color represents the video category, and blue is the target category to count the number of effective views. (b) The trend of EVTR with increasing consumption of videos with the same category. We include the EVTR of the video with the target category (same) and videos with other categories close to it (other).

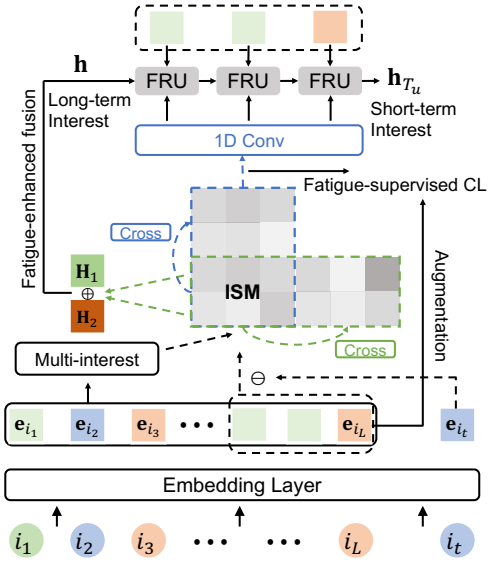


Figure 2: The framework of FRec.

Input: Historical sequences for all the users $\{S_u | u \in \mathcal{U}\}$.
Output: A model that can predict the user's interaction probability for the next (target) item i_t.

3 METHOD

Figure 2 shows the framework of our FRec with four modules.

3.1 Interest-aware Similarity Matrix

Fine-grained target-historical item similarity is necessary to support the modeling of user fatigue. Indeed, the similarity between two items can stem from multiple aspects and correspond to multiple sub-interests of the user. For instance, the similarity of videos can be characterized by aspects such as shooting style, video tone, and topics, all of which can be used to model user fatigue when watching videos. Therefore, we first extract multi-interests from historical sequences.

Table 1: Frequently used notations.

S_u, L_u	Historical sequence for user u, and its length.
$\hat{S}_u$, T_u	Sub-sequence with recent items in S_u, and its length.
T	Truncated threshold for selecting sub-sequence $\hat{S}_u$.
K	The number of interests.
C	The number of cross and convolutional layers.
$\mathbf{M}_j (\mathbf{M}_j^\top)$	The j-th column of the matrix $\mathbf{M}$ (transposed $\mathbf{M}^\top$).
$\mathbf{e}_i$	The embedding of item i.
$\mathbf{H}$	Multi-interest embedding matrix.
$\mathbf{F}$	Interest-aware similarity matrix.
$\mathbf{h}, \mathbf{h}_{T_u}$	Long and short-term interest embedding.
$\mathbf{MLP}$	Multi-layer perceptron applied on the last dimension.
$\mathbf{W}, \mathbf{b}$	Learnable weight matrix and bias vector.

First of all, each item i is assigned an embedding $\mathbf{e}_i \in \mathbb{R}^{d \times 1}$, where d is the embedding dimension. Correspondingly, the sequence S_u for user u can be encoded as an embedding matrix $\mathbf{S}_u \in \mathbb{R}^{d \times L_u}$, where the l-th column is $\mathbf{e}_{i_l}$, the embedding for item i_l in the sequence. We then choose a widely-used self-attention mechanism [5, 28] for multi-interest extraction. Specifically, we generate a multi-interest embedding matrix for user u as follows,

$$\begin{aligned} \mathbf{H} &= \mathbf{S}_u \mathbf{A}, \\ \mathbf{A} &= softmax(\mathbf{MLP}_1(\mathbf{S}_u^\top)), \end{aligned} \tag{1}$$

where $\mathbf{MLP}_1$ is a two-layer perceptron with *tanh* as nonlinear activation, and the output dimension is the number of interests K, which is a tunable hyper-parameter. Here $\mathbf{A} \in \mathbb{R}^{L_u \times K}$ is attention weights for aggregating all the item embeddings in the sequence, which is generated by applying *softmax* along with the first dimension of $\mathbf{MLP}_1$ output. Finally, we obtain K interest embeddings $\mathbf{H} \in \mathbb{R}^{d \times K}$ from the user's historical interactions.

To obtain fine-grained target-historical similarity, we leverage extracted multi-interest and item embeddings in latent space. Compared with existing works utilizing coarse-grained item-level or category-level features [29, 38], the embedding-based similarity can measure relevance more accurately and effectively. Specifically, we construct an interest-aware similarity matrix to measure the similarity between the target item i_t and historical item i_l with respect to each user interest, formulated as follows,

$$\mathbf{F}_{l,k} = \frac{1}{1 + \left| \frac{\mathbf{e}_{i_t}^\top \mathbf{H}_k}{\|\mathbf{H}_k\|} - \frac{\mathbf{e}_{i_l}^\top \mathbf{H}_k}{\|\mathbf{H}_k\|} \right|}, \tag{2}$$

where the similarity is based on the projection distance between the embedding of i_t and i_l on the k-th interest embedding $\mathbf{H}_k$, and shorter distance means higher similarity. Figure 3 illustrates how this similarity is calculated. Considering that user fatigue is the most relevant to items nearest to the target item, we confine the calculation of this similarity feature among the most recent $T_u = \min(T, L_u)$ items, *i.e.*, $l \in \{L_u - T_u + 1, L_u - T_u + 2, \cdots, L_u\}$. T is a tunable truncated threshold to control how many recent items should be included. We denote this sub-sequence of items as $\hat{S}_u$.

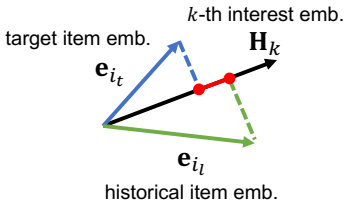


Figure 3: Illustration of the calculation of interest-aware similarity, represented by the length of the red line.

The similarity matrix $\mathbf{F} \in \mathbb{R}^{T\times K}$ will be padded with zeros as $\mathbf{F}^\top = [\mathbf{F}^\top, \mathbf{0}^\top_{K\times(T-L_u)}]$ if $T > L_u$, where $[\cdot,\cdot]$ denotes the concatenation operation along the last dimension. It will support the modeling of user fatigue along with capturing users' long and short-term interests.

3.2 Fatigue-enhanced Multi-interest Fusion

Although multi-interest embeddings $\mathbf{H}$ capture multiple aspects of interests, there is a critical influence of user fatigue on long-term interests with respect to the target item. In other words, the importance should be decreased if the user is experiencing fatigue for a certain sub-interest. To adaptively adjust long-term interest, we propose a fatigue-enhanced multi-interest fusion built upon interest-aware similarity matrix $\mathbf{F}$. A direct way is,

$$\begin{aligned} \mathbf{h} &= \mathbf{H}\mathbf{w}, \\ \mathbf{w} &= softmax(\mathbf{MLP}_2(\mathbf{F}^\top)), \end{aligned} \tag{3}$$

where $\mathbf{MLP}_2$ has the output dimension of 1, and attention weights $\mathbf{w} \in \mathbb{R}^{K\times 1}$ for interests fusion are obtained from similarity features with respect to each sub-interest. However, there are two key difficulties when learning fatigue-aware importance. On the one hand, the dependency of user fatigue on target-historical similarity can be nonlinear or even more complex [1, 30], and directly feeding these features into neural networks may not guarantee accurate modeling. On the other hand, several recently consumed items can jointly contribute to user fatigue on the target item. For example, compared with consuming only one video, a user experience much more fatigue if he/she has consumed five videos published by the same author of the target item within one hour. To tackle these problems, inspired by [37], we propose to utilize feature cross and apply C layers of Cross Network on the similarity matrix. Specifically, each cross layer processes the c-th features as follows,

$$\mathbf{P}_{c+1} = \mathbf{P}_0 \odot (\mathbf{W}_c\mathbf{P}_c) + \mathbf{P}_c, \tag{4}$$

where $\mathbf{W}_c \in \mathbb{R}^{T\times T}$ is a learnable weight matrix and $\odot$ denotes element-wise product. The layer c ranges from 0 to $C-1$ and $\mathbf{P}_0 = \mathbf{F}$. In this way, the feature interplay of the same item can generate high-order features for modeling complex similarity-fatigue dependency. The interplay between different items can assist in modeling the effect of multiple items on user fatigue. Finally, Eq. 3 can be modified as follows,

$$\begin{aligned} \mathbf{h} &= \mathbf{H}\mathbf{w}, \\ \mathbf{w} &= softmax\left(\mathbf{MLP}_2\left([\mathbf{P}_C^\top, \mathbf{P}_0^\top]\right)\right). \end{aligned} \tag{5}$$

With the fatigue-enhanced fusion, we obtain the user's long-term interest embedding $\mathbf{h} \in \mathbb{R}^{d\times 1}$.

3.3 Fatigue-gated Recurrent Unit

As stated in the previous subsection, recently consumed items in the historical sequence are important in causing user fatigue. Therefore, we model the influence of temporal user fatigue on short-term interest learning. Similarly, the similarity features are first processed with cross networks to tackle the problems of complex similarity-fatigue dependency and joint effects from multiple interests. Specifically, C cross layers are applied as follows,

$$\mathbf{Q}_{c+1} = \mathbf{Q}_0 \odot (\mathbf{Q}_c\mathbf{W}'_c) + \mathbf{Q}_c, \tag{6}$$

where $\mathbf{W}'_c \in \mathbb{R}^{K\times K}$ is a learnable weight matrix and $\mathbf{Q}_0 = \mathbf{F}$. Furthermore, the temporal pattern contained in the sequence of recent items is also necessary for modeling fatigue. For instance, consuming five consecutive items similar to the target item will cause a more heightened perception of fatigue than that of disordered ones. Inspired by the effectiveness of CNNs in modeling sequences [4, 35], we apply 1D convolutional networks to further model temporal user fatigue sequentially. Each layer of convolution operation for the l-th item in the sub-sequence $\hat{S}_u$ is formulated as follows,

$$\begin{aligned} \hat{\mathbf{Q}}_l^\top &= [q_l^1, q_l^2, \cdots, q_l^{d_{\text{out}}}]^\top, \\ q_l^n &= LeakyRelu\left(\text{SUM}\left(\hat{\mathbf{Q}}^\top_{l-s+1:l} \odot \mathbf{W}^n_{\text{conv}}\right)\right), \end{aligned} \tag{7}$$

where $\mathbf{W}^n_{\text{conv}} \in \mathbb{R}^{d_{\text{in}}\times s}$ is the n-th learnable filter kernel, and d_{in} and s denotes the input dimension and the kernel size respectively. The number of filter kernels (*i.e.*, the output dimension of this convolutional layer) is d_{out}. The input is the crossed features obtained above, *i.e.*, the initial $\hat{\mathbf{Q}} = [\mathbf{Q}_C, \mathbf{Q}_0]$, thus the initial $d_{\text{in}} = 2K$. After C layers of convolution, we model temporal fatigue until the l-th item in the representation $\hat{\mathbf{Q}}_l$. Note that we use 'causal' convolutions [4] since current fatigue only depends on previous items. Zero padding is utilized when $l < s$.

In terms of modeling short-term interest, RNNs have been demonstrated as effective modules in many advanced works [16, 42–44], such as GRU, LSTM, *etc.* To incorporate fatigue influence, we propose a fatigue-gated recurrent unit (FRU) built upon GRU. Specifically, the extracted fatigue representation until each item serves as additional feature input to construct update and reset gates. With the state input $\mathbf{h}_{l-1} \in \mathbb{R}^{d_{\text{in}}\times 1}$ from previous step and embedding input $\mathbf{x}_l \in \mathbb{R}^{d_{\text{in}}\times 1}$, new state $\mathbf{h}_l \in \mathbb{R}^{d_{\text{out}}\times 1}$ is calculated as follows,

$$\begin{aligned} \mathbf{z}_l &= sigmoid(\mathbf{W}_z\mathbf{x}_l + \mathbf{U}_z\mathbf{h}_{l-1} + \underline{\mathbf{V}_z\hat{\mathbf{Q}}_l} + \mathbf{b}_z), \\ \mathbf{r}_l &= sigmoid(\mathbf{W}_r\mathbf{x}_l + \mathbf{U}_r\mathbf{h}_{l-1} + \underline{\mathbf{V}_r\hat{\mathbf{Q}}_l} + \mathbf{b}_r), \\ \hat{\mathbf{h}}_l &= tanh\left(\mathbf{W}_h\mathbf{x}_l + \mathbf{U}_h(\mathbf{r}_l \odot \mathbf{h}_{l-1}) + \mathbf{b}_h\right), \\ \mathbf{h}_l &= (1-\mathbf{z}_l) \odot \mathbf{h}_{l-1} + \mathbf{z}_l \odot \hat{\mathbf{h}}_l, \end{aligned} \tag{8}$$

where $\mathbf{W}_{z,r,h}, \mathbf{U}_{z,r,h}, \mathbf{V}_{z,r} \in \mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}$ and $\mathbf{b}_{z,r,h} \in \mathbb{R}^{d_{\text{out}}\times 1}$ are learnable weights and bias. The embedding input is the l-th item's embedding, *i.e.*, $\mathbf{x}_l = \mathbf{e}_{i_l}$. We set the initial state $\mathbf{h}_0 = \mathbf{h}$, which is the long-term interest embedding with fatigue-enhanced fusion. In this formula, temporal user fatigue affects how short-term interests evolve. Generally speaking, the interests before the current time step should not be propagated to the next step if the corresponding

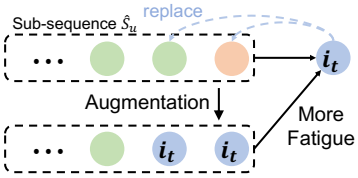


Figure 4: Illustration of the sequence augmentation to obtain fatigue signals. There is more fatigue if some items in the sub-sequence $\hat{S}_u$ are replaced by the target item.

fatigue is intense. We apply FRU on the sub-sequence $\hat{S}_u$, and the final output $\mathbf{h}_{T_u} \in \mathbb{R}^{d_{\text{out}} \times 1}$ encodes user's short-term interests with fatigue influence.

3.4 Fatigue-supervised Contrastive Learning

Although we have encoded temporal fatigue in latent space, whether these representations guarantee the modeling of real fatigue is unknown. The challenge is that there are no explicit signals for the supervision of representation learning. Inspired by the advantages of self-supervised learning [41] and multi-task learning [32] in recommendations, we propose a novel sequence augmentation for fatigue-supervised contrastive learning. Specifically, $N \in [\max(N_r, 1), T_u]$ items in the sub-sequence $\hat{S}_u$ are replaced by the target item, where N_r is the number of repetitions of the target item in $\hat{S}_u$. As shown in Figure 4, the primary idea is that users will experience more fatigue if they have much more repetitive consumption. We set a margin N_r when choosing how many items to replace, which is large enough from the experimental results.

With the augmented sequence, we can also obtain similarity features $\mathbf{Q}'_0$ (and the processed $\mathbf{Q}'_C$ by cross networks) motivating the modeling of temporal fatigue after the same modeling introduced in previous subsections. All the learnable parameters are shared when modeling original and augmented sequences. Finally, user fatigue for the interaction of the target item can be predicted as follows,

$$\begin{aligned} f &= \text{MEAN}\left(\mathbf{MLP}_3\left([\mathbf{Q}_C, \mathbf{Q}_0]\right)\right), \\ f' &= \text{MEAN}\left(\mathbf{MLP}_3\left([\mathbf{Q}'_C, \mathbf{Q}'_0]\right)\right). \end{aligned} \tag{9}$$

All the items in $\hat{S}_u$ are considered for modeling the fatigue with MEAN. We then formulate the contrastive loss with fatigue as the supervision as follows,

$$\mathcal{L}_{\text{con}} = \sum -\log \frac{\exp(-f)}{\exp(-f) + \sum_{j=1}^{4} \exp(-f'_j)}, \tag{10}$$

where f'_j denotes the fatigue of the j-th augmentation, and we conduct four augmentations for each instance. Note that the fatigue should be larger after the augmentation, thus we use $-f$ to calculate the likelihood.

3.5 Model Training

We consider both users' long-term and short-term interests for interaction predictions, as well as user fatigue when making the decision on the target item. The prediction score for the user u and the target items i_t is calculated as follows,

$$y_{u,i_t} = \mathbf{MLP}_4([\mathbf{h}^\top, \mathbf{h}_{T_u}^\top, \mathbf{e}_{i_t}^\top]) - tanh(f). \tag{11}$$

We explicitly decrease the score with predicted user fatigue obtained from the subsection above. The function *tanh* controls the magnitude of effects.

The recommendation loss for model training is a widely-used softmax loss function [5], formulated as follows,

$$\mathcal{L}_{\text{rec}} = \sum_{(u,i_t,i'_1 \sim i'_4) \in O} -\log \frac{\exp(y_{u,i_t})}{\exp(y_{u,i_t}) + \sum_{j=1}^{4} \exp(y_{u,i'_j})}, \tag{12}$$

where O denotes all the training data, and $i'_1 \sim i'_4$ are randomly sampled negative items for each (u, i_t) pair.

Final training loss is the combination of recommendation and contrastive loss,

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \alpha \mathcal{L}_{\text{con}}, \tag{13}$$

where α is a hyper-parameter for controlling the importance of fatigue supervision.

4 EXPERIMENTS

To evaluate our proposed FRec, we conduct extensive experiments on both public and large-scale industrial datasets. We will answer the following research questions in this section,

- **RQ1**: Can FRec outperform state-of-the-art models in terms of recommendation accuracy?
- **RQ2**: Can each key module benefit the overall performance?
- **RQ3**: Can FRec result in the reduction of user fatigue?
- **RQ4**: How do key hyper-parameters influence the performance?

4.1 Experimental Settings

Datasets. The statistics of datasets are shown in Table 2, where Avg. Length denotes the sequence length averaged over all the users.

- **Kuaishou**[2]. This is one of the largest micro-video platforms in China and this dataset has been used in many related works [6, 43]. It contains users' interactions with micro-videos over one week (October 22 to October 28, 2020), and records various behaviors such as click, like, follow, *etc.* We extract the click interactions for experiments.
- **Taobao**[3]. This is the largest e-commerce platform in China. This dataset records users' interactions with various products from November 25 to December 3, 2017, including page view, cart, purchase, *etc.* We follow existing works [43] and choose the data of page view for experiments.
- **Industrial**. The interaction data is collected from Kuaishou for 1 hour, involving tens of millions of users. Unlike public Kuaishou dataset, we include various behavioral data for experiments to model user fatigue from uninterrupted behavioral sequences.

We adopt widely-used 10-core rules [6, 43] for public datasets to filter out inactive users and unpopular items. We split sequential interactions chronologically into 8:1:1 for training, validating, and testing models [5]. Since we model user fatigue with respect to both short-term and long-term interests (such as a period of several days), the maximum sequence length is set longer than average length, *i.e.*, 250 for Kuaishou dataset and 100 for Taobao dataset.

[2]https://www.kuaishou.com
[3]https://tianchi.aliyun.com/dataset/dataDetail?dataId=649

Table 2: Statistics of three datasets.

Dataset	#Users	#Items	#Instances	Avg. Length
Kuaishou	37,502	131,063	6,427,764	171.4
Taobao	41,101	90,524	2,256,967	54.9
Industrial	38,467,817	19,863,454	804,934,827	20.9

Baselines. We choose the following state-of-the-art (SOTA) recommendation models for comparisons, 1) long-term and (or) short-term interest modeling: **DIN** [45], **DIEN** [44], **GRU4Rec** [16], **SASRec** [19], **AdaMCT** [18], **Caser** [35], **SLi-Rec** [42], and **CLSR** [43], 2) multi-interest modeling: **SUM** [26], **ComiRec** [5] with two versions of extracting multiple interests by dynamic routing (-DR) or self-attention (-SA), and **MGNM** [36], 3) fatigue modeling[4]: **DFN** [38].

Evaluation Metrics. Similar to existing works sampling one negative item for each positive instance [6, 42, 43], we sample nine negative items to ensure robust training and evaluation [27]. We adopt widely-used accuracy metrics AUC and GAUC [45] as well as ranking metrics HR@k, NDCG@k, and MRR [6, 27, 36, 43] for performance evaluation. We set k as 2 and 4, a widely-used setting in existing works [6, 43].

Hyper-parameter Settings. We implement our FRec and all the baselines with Microsoft Recommender[5] based on Tensorflow[6]. We use Adam [20] optimizer for modeling learning, where the initial learning rate is 0.001. L2 regularization weight is searched among {1e-4, 1e-6}. The batch size for training is set as 500. We early stop the training process when GAUC on the validation set decreases for two consecutive epochs. Embedding dimension d is set as 40 for all the models. The $\mathbf{MLP}_4$ for final prediction is three-layer with hidden size [100, 64], with *relu* as activation function, and batch normalization. We conduct a careful grid search to find optimal hyper-parameters for each model, following the original papers. For our FRec, the kernel size of convolutional layers $s = 5$, the number of interests $K = 4$, and the number of cross and convolutional layers $C = 2$. For the convolution, the number of filter kernels (*i.e.*, hidden dimension) is [20, 40]. In FRU, $d_{in} = d_{out} = 40$. Regarding other $\mathbf{MLP}_{1,2,3}$, they are two-layer with the hidden size as half of the input dimension. The truncated threshold T is set as 50 for Kuaishou and 40 for the Taobao dataset. The weight of contrastive learning $\alpha = 0.4$. Note that these hyper-parameters are not carefully tuned for better performance.

4.2 Overall Comparison (RQ1)

Public Datasets. The performance on the Kuaishou and Taobao datasets are shown in Table 3. From the comparison, we have the following observations,

- **FRec outperforms all the baselines significantly.** On the Kuaishou dataset, FRec improves by about 0.009 in terms of AUC and GAUC. Corresponding improvements are 0.026 and 0.019 on the Taobao dataset. For other ranking metrics, the improvements range from 1.3%∼3.0% and 2.4%∼ 5.8% on the Kuaishou and Taobao datasets, respectively. The p-value < 0.001 demonstrates that FRec can give consistently and significantly more accurate recommendations than SOTA models.
- **Modeling long and short-term interests or multi-interests can obtain better performance generally.** CLSR, a SOTA model disentangling users' long and short-term interests based on causal structure, obtains almost the best performance among the baselines on both datasets. Compared with models only capturing long-term (*e.g.*, DIN) or short-term (*e.g.*, GRU4Rec) interest, jointly modeling (CLSR, SLi-Rec) is better on most metrics. ComiRec-SA, a multi-interest framework based on self-attention, also obtains competitive performance.
- **Feeding coarse-grained similarity features can benefit model performance.** On both datasets, DFN outperforms the backbone DIN on most metrics, especially on AUC. However, top performance can not be guaranteed since it directly concatenates several fatigue-aware features (*e.g.*, the number of historical consumed items, the number of items that belong to the same category with the target item) with the embedding of the target item. In other words, it's necessary to capture complex influence of user fatigue on interests accurately in model design.

Industrial Dataset. For the industrial deployment on Kuaishou, we select these baseline methods performing well on the public Kuaishou dataset. Specifically, the baseline methods and our method are deployed to the click-through rate (CTR) prediction module in the industrial recommendation engine. The results are shown in Table 4. Our FRec can improve both AUC and GAUC by more than 0.01 compared with the best baselines. This is a huge improvement for a real-world scenario with tens of millions of users [8, 14]. Note that this dataset has been scaled up hundreds of times, thus the improvement is more promising compared with that of the public Kuaishou dataset. We attribute this advantage to the choice of negative items in the evaluation. On the public Kuaishou dataset, negative items are randomly sampled. In contrast, we use the exposed videos users have not clicked on industrial dataset. In this scenario, user fatigue plays a critical role in consecutive decisions, where the exposed videos have generally matched user interests guaranteed by the advanced industrial recommender system. Therefore, FRec can effectively distinguish between clicked and non-clicked ones among all the exposed videos when modeling the influence of user fatigue on temporal interests accurately.

Efficiency Comparison. Table 5 shows the training time per epoch of all the models on public datasets, demonstrating that the efficiency of FRec is comparable with simple and complex baselines.

4.3 Ablation Study (RQ2)

In our proposed FRec, there are some key modules for modeling user fatigue, including 1) fatigue-enhanced multi-interests fusion, 2) fatigue recurrent unit (FRU) with fatigue representations as additional input, 3) cross networks for feature interplay to handle

[4]Although there is another method modeling user fatigue for click-through rate prediction [24], we don't include it because modeling fatigue relies on *non-click* historical sequences and rich context features. Besides, it obtains supervision signals of user fatigue through interaction data in future three days. These are unusual settings and not applicable to our general problem.

[5]https://github.com/microsoft/recommenders

[6]https://www.tensorflow.org

[7]Due to the requirement of constructing graphs based on historical sequences, MGNM encounters out-of-memory (OOM) on the Kuaishou dataset, which has a longer maximum sequence length of 250. In original paper, the maximum sequence length is only 100.

Table 3: Performance comparison on public dataset. All the results are averaged over five experiments. Underline means the best two baselines, and bold means p-value < 0.001 compared with the best baseline under the student's t-test.

	Model	DIN	DIEN	GRU4Rec	SASRec	AdaMCT	Caser	SLi-Rec	CLSR	SUM	ComiRec -DR	ComiRec -SA	MGNM	DFN	FRec
Kuaishou	AUC	0.6054	0.7520	0.8306	0.8298	0.8067	0.8228	0.8258	0.8263	0.8235	0.8239	0.8441	OOM[7]	0.6613	**0.8533**
	GAUC	0.8204	0.8198	0.8401	0.8270	0.8033	0.8417	0.8388	0.8473	0.8414	0.8259	0.8464	OOM	0.8159	**0.8564**
	HR@2	0.6179	0.6249	0.6570	0.6226	0.5776	0.6552	0.6651	0.6703	0.6570	0.6301	0.6658	OOM	0.6284	**0.6878**
	HR@4	0.8269	0.8356	0.8642	0.8466	0.8172	0.8683	0.8585	0.8747	0.8670	0.8429	0.8705	OOM	0.8424	**0.8860**
	NDCG@2	0.5417	0.5484	0.5784	0.5428	0.4982	0.5749	0.5897	0.5901	0.5779	0.5523	0.5869	OOM	0.5509	**0.6077**
	NDCG@4	0.6403	0.6479	0.6765	0.6486	0.6112	0.6758	0.6812	0.6869	0.6772	0.6527	0.6837	OOM	0.6519	**0.7016**
	MRR	0.6045	0.6111	0.6355	0.6073	0.5719	0.6327	0.6442	0.6444	0.6353	0.6143	0.6422	OOM	0.6136	**0.6583**
Taobao	AUC	0.6800	0.7592	0.8257	0.8455	0.8412	0.8264	0.8333	0.8527	0.8247	0.7820	0.8359	0.7291	0.7630	**0.8795**
	GAUC	0.8469	0.8263	0.8327	0.8430	0.8336	0.8376	0.8381	0.8601	0.8281	0.7779	0.8333	0.7279	0.8459	**0.8792**
	HR@2	0.7072	0.6737	0.6922	0.6964	0.6842	0.6878	0.6857	0.7305	0.6818	0.5675	0.6667	0.4897	0.7144	**0.7660**
	HR@4	0.8585	0.8393	0.8331	0.8460	0.8325	0.8417	0.8464	0.8667	0.8312	0.7702	0.8374	0.7055	0.8485	**0.8873**
	NDCG@2	0.6444	0.6101	0.6397	0.6373	0.6268	0.6311	0.6224	0.6754	0.6248	0.5010	0.6039	0.4258	0.6631	**0.7143**
	NDCG@4	0.7159	0.6883	0.7061	0.7079	0.6967	0.7036	0.6983	0.7397	0.6953	0.5964	0.6845	0.5271	0.7263	**0.7716**
	MRR	0.6897	0.6623	0.6888	0.6851	0.6765	0.6818	0.6723	0.7177	0.6752	0.5736	0.6585	0.5121	0.7082	**0.7501**

Table 4: Performance comparison on the industrial dataset.

Metric	GRU4Rec	SLi-Rec	CLSR	ComiRec-SA	FRec
AUC	0.7252	0.7302	0.7267	0.7247	**0.7408**
GAUC	0.6525	0.6604	0.6584	0.6433	**0.6709**

complex fatigue influence on user interests, and 4) contrastive learning with explicit fatigue supervision. To extensively verify their benefits, we conduct ablation studies to investigate how each module influences model effectiveness. Correspondingly, we have made the following changes,

- **w/o Fusion**: replace fatigue-enhanced attentive fusion with mean pooling.
- **w/o FRU**: replace FRU with vanilla GRU, with convolutional fatigue features removed.
- **w/o Cross**: replace cross layers with dense layers.
- **w/o CL**: remove contrastive learning, *i.e.*, set $\alpha = 0$.

Performances with these key modules removed are shown in Figure 5. Results show that FRec obtains better performances consistently compared with all the incomplete models on both datasets, demonstrating the necessity of the proposed modules for modeling user fatigue. Note that significantly worse performance without cross networks indicates critical benefits of the interplay of similarity features. Furthermore, on the Kuaishou dataset, the performance drops the most when contrastive learning is removed, but this is not the case for the Taobao dataset. We attribute this difference to the effectiveness of sequence augmentation for obtaining fatigue signals. In Kuaishou, a micro-video platform, repetitive recommendations of the same videos obviously cause intense user fatigue. However, for e-commerce platforms like Taobao, multiple views of the same product page are relatively common, and users may not experience fatigue during limited repetitions. In contrast, FRU plays an essential role in the model on the Taobao dataset. This demonstrates the effectiveness of guiding short-term interest evolution with temporal user fatigue as necessary inputs for update and reset gates. It can be explained by the intention changes of sequential behaviors in the e-commerce applications [7, 25]. Specifically, users have different intentions when browsing products, and they may experience fatigue of redundant exposure if the intention has switched. Therefore, fusing temporal user fatigue can assist in modeling short-term interests more accurately.

4.4 Study on Fatigue Reduction (RQ3)

Online Experiments. We further deploy FRec on Kuaishou to verify the effectiveness of fatigue reduction and satisfaction improvement. Specifically, we choose CLSR (the highest overall performance on public and industry datasets) for comparison and conduct an A/B test for 7 days, involving millions of users. Table 6 shows the improvement of key online metrics, which are defined as follows,

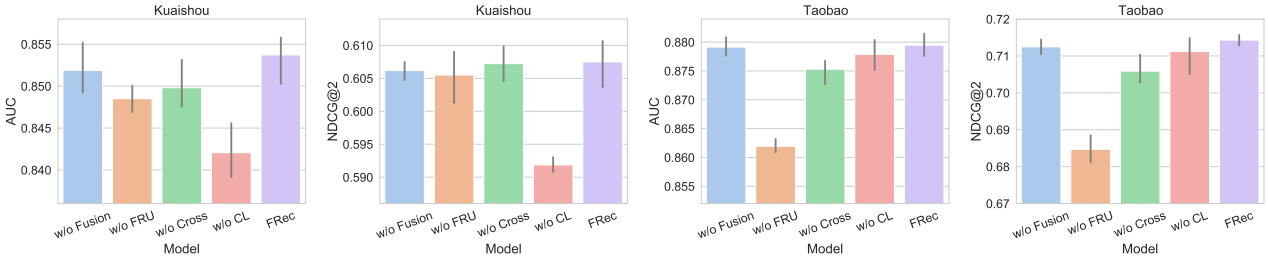
- **App usage** denotes the average dwell time users use the App.
- **#Play** denotes total number of effectively played videos.
- **#Category** denotes the average number of video categories in terms of the behavior of effective view.
- **Concentration** indicates how similar videos consecutively exposed over a fixed window are. It's calculated as $N - C$, where $N = 6$ denotes the number of videos in the window, and C is the number of video categories.

FRec improves all the metrics significantly around 0.1%~0.4%, which is impressive for large-scale online experiments. Obviously, FRec not only enables users to spend more time on the platform and view more videos but also promotes a more diverse video consumption. Particularly, the reduction in **Concentration** indicates that there are fewer videos of the same category in consecutive exposures of a short period, which lowers the perception of user fatigue.

In order to further demonstrate the effectiveness of fatigue reduction, we conduct a similar analysis in Figure 1 with online interaction data. The comparison between the results of CLSR and FRec is shown in Figure 8, where EVTR is also normalized. It's obvious that FRec can improve EVTR significantly when users have consumed many similar videos. Note that the improvement is obtained in an online setting, thus this is strong evidence of fatigue reduction.

Table 5: Training time (minutes) per epoch of all the models.

Model	DIN	DIEN	GRU4Rec	SASRec	AdaMCT	Caser	SLi-Rec	CLSR	SUM	ComiRec -DR	ComiRec -SA	MGNM	DFN	FRec
Kuaishou	17.0	17.2	18.8	59.3	17.8	16.8	24.1	21.7	83.2	16.6	17.0	OOM	19.5	23.2
Taobao	7.8	8.5	9.8	14.0	9.0	13.4	11.1	11.3	35.3	7.9	7.9	30.0	10.0	12.7



(a) AUC on Kuaishou (b) NDCG@2 on Kuaishou (c) AUC on Taobao (d) NDCG@2 on Taobao

Figure 5: Ablation study of key modules. Error bar denotes 95% confidence intervals for five experiments.

Offline Experiments. Since it's not practical to conduct similar experiments on offline datasets, we conduct alternative experiments to demonstrate the fatigue reduction of FRec. We first show a recommendation case for a user in the industrial dataset. As illustrated in Figure 6, the user has watched many sports videos recently, but SLi-Rec (the best baseline) still ranks a basketball video in the first position. In contrast, our FRec ranks this video in the last position. This demonstrates that FRec can alleviate improper and repetitive recommendations that may cause user fatigue in a period of short time (*e.g.*, five minutes). Besides, as a part of user interests, the video about the scenery and trips lies at the top, with which the user has limited interactions. This demonstrates that FRec can assist in satisfying user interests and reducing user fatigue adaptively.

Furthermore, we investigate whether FRec can perform better when user fatigue plays an important role in user decisions. Specifically, we define a proxy to represent the importance, formulated as follows,

$$m = \sum_{i_n} (m_{i_n} - m_{i_p}), \tag{14}$$

where m_{i_n} (m_{i_p}) denotes the number of items within three-hour historical consumption that belong to the same category of the negative (positive) item i_n (i_p). This proxy m indicates the difference between historical-negative and historical-positive item similarity. High $m > 0$ means that only modeling user interests based on relevance learning is insufficient for accurate recommendations. In other words, users are willing to interact with the item i_p other than i_n because of experiencing fatigue. We divide all the instances into groups with different m, and show the performance comparison in Figure 7. Due to the sparsity of m on the Kuaishou dataset, we only report results on the Taobao dataset. The best two baselines perform worse with increasing m, especially when $m \geq 5$. In contrast, our FRec keeps steady (and significantly better) performances thanks to the ability to model users' temporal fatigue in short-term interest learning. Therefore, FRec can reduce user fatigue by ranking positive items at the top, which are less similar to historical items than negative items.

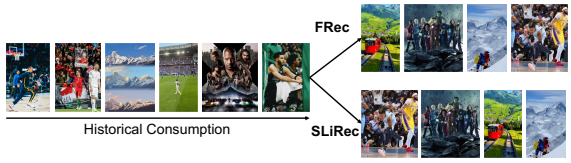
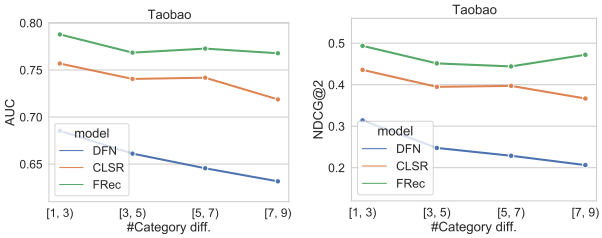


Figure 6: Recommendations by SLiRec and FRec.



(a) AUC on Taobao (b) NDCG@2 on Taobao

Figure 7: Performance of instance groups with different m.

Table 6: The improvement of key online metrics. ↑ (↓) means higher (lower) is better.

Metric	App usage ↑	#Play ↑	#Category ↑	Concentration ↓
Impr. (%)	+0.300	+0.466	+0.408	-0.136

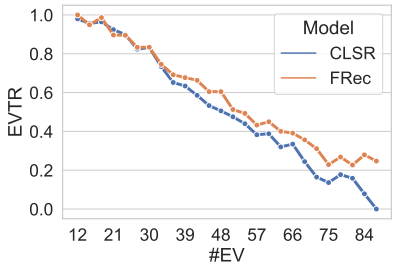
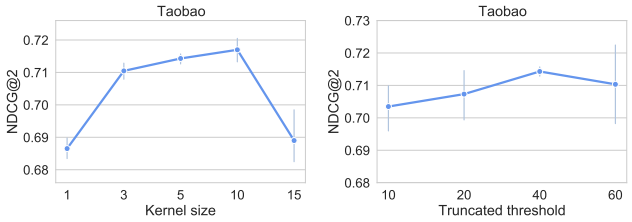


Figure 8: Online EVTR comparison between CLSR and FRec.



(a) Kernel size (b) Truncated threshold

Figure 9: Impact of kernel size and truncated threshold in 1D convolution. Error bar denotes 95% confidence intervals for five experiments.

4.5 Hyper-parameter Study (RQ4)

We further investigate how key hyper-parameters impact the effectiveness of modeling user fatigue and recommendation performances. Due to the page limitations, we only report the results on the Taobao dataset. The results on the Kuaishou dataset imply the same conclusions.

Kernel size. As stated in subsection 3.3, considering consecutive items is necessary for modeling temporal fatigue. Therefore, we compare the performances under different kernel sizes of convolutions to verify whether FRec captures this pattern. As shown in Figure 9 (a), the performance is very low when kernel size is 1 and is higher with a larger size. This is direct evidence that FRec can effectively model user fatigue caused by consecutive consumption. Note that the performance also drops when the size is too large, this may be explained by users' limited memory of historical experience.

Truncated threshold. From the results shown in Figure 9 (b), we observe that using too short (10 items) or too long (60 items) subsequence both leads to worse performances. This can be attributed to the missing of critical items and the inclusion of noisy items respectively, when modeling short-term interests and temporal user fatigue. Besides, from the comparison of confidence intervals, we conclude that the proper choice of the truncated threshold assists in obtaining stable recommendation results.

5 RELATED WORK

5.1 Sequential Recommendation

In recent years, deep learning has been widely applied to sequential recommendations, and many advanced neural networks have been utilized for modeling long and short-term interests [3, 22, 23, 34]. Specifically, RNNs can directly capture the evolution of users' short-term interests when encoding sequential items, including gated recurrent unit (GRU) [9], long short-term memory (LSTM) [17], *etc.* Similarly, CNNs have also been exploited to learn temporal patterns in historical consumption [35, 39, 40]. In terms of long-term interests, some works rely on matrix factorization or attention mechanisms [22, 23, 42], which requires taking the whole sequence into consideration simultaneously. Recently, these two aspects have been disentangled for better modeling from the perspective of causal inference [43]. Many works also propose to encode users' multiple interests by multiple representations simultaneously. In general, there are three types of modules for generating multiple interest representations, including multi-channel memory networks [26, 31], dynamic routing [5, 21], and self-attention mechanism [5].

Different from these works, we take user fatigue into consideration and model its influence on long and short-term interests.

5.2 User Fatigue in Recommendation

Modeling user fatigue has not received much attention from the academic community of the recommender system. Several existing works [1, 24, 29, 30, 38] rely on coarse-grained features representing how similar the target item is to historical consumption, such as how many historical items belong to the same category of the target item, *etc.* Then these features are directly feeded into the base recommender (*e.g.*, decision trees) [29, 38] or utilized for modeling fatigue by a quadratic function [30]. On one hand, these methods require manual feature engineering and enough features are difficult to obtain when relevant data is missing. On the other hand, the way to model user fatigue is not carefully designed to handle complex relationships with similarity features.

In this work, we leverage fine-grained similarity features to support the modeling of user fatigue. Besides, user fatigue is also explicitly predicted based on contrastive learning.

6 CONCLUSION AND FUTURE WORK

In this work, we propose to model user fatigue in interest learning for sequential recommendations. Specifically, based on a multi-interest framework, we develop an interest-aware similarity matrix for fatigue modeling and handle its influence on long and short-term user interests. We also propose a novel sequence augmentation to obtain fatigue signals as supervision for contrastive learning. Extensive offline and online experiments demonstrate the effectiveness of our model in improving user experience and reducing user fatigue. As for future work, we will propose introducing a fatigue metric as a new dimension to explicitly measure the effectiveness of recommendations, thereby encouraging the development of fatigue modeling in recommender system research.

ACKNOWLEDGEMENT

This work is supported by the National Natural Science Foundation of China under U23B2030, 62272262 and 72342032. This work is supported by a grant from the Guoqiang Institute, Tsinghua University under 2021GQG1005. This work is also supported by Kuaishou.

REFERENCES

[1] Michal Aharon, Yohay Kaplan, Rina Levy, Oren Somekh, Ayelet Blanc, Neetai Eshel, Avi Shahar, Assaf Singer, and Alex Zlotnik. 2019. Soft frequency capping for improved ad click prediction in yahoo gemini native. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2793–2801.

[2] Bushra Alhijawi, Arafat Awajan, and Salam Fraihat. 2022. Survey on the objectives of recommender systems: Measures, solutions, evaluation methodology, and new perspectives. *Comput. Surveys* 55, 5 (2022), 1–38.

[3] Mingxiao An, Fangzhao Wu, Chuhan Wu, Kun Zhang, Zheng Liu, and Xing Xie. 2019. Neural news recommendation with long-and short-term user representations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 336–345.

[4] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271* (2018).

[5] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Controllable multi-interest framework for recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2942–2951.

[6] Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2021. Sequential Recommendation with Graph Neural Networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 378–387.

[7] Yongjun Chen, Zhiwei Liu, Jia Li, Julian McAuley, and Caiming Xiong. 2022. Intent contrastive learning for sequential recommendation. In *Proceedings of the ACM Web Conference 2022*. 2172–2182.

[8] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. 7–10.

[9] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[10] Jingtao Ding, Yuhan Quan, Xiangnan He, Yong Li, and Depeng Jin. 2019. Reinforced Negative Sampling for Recommendation with Exposure Data.. In *IJCAI*. Macao, 2230–2236.

[11] Jingtao Ding, Yuhan Quan, Quanming Yao, Yong Li, and Depeng Jin. 2020. Simplify and robustify negative sampling for implicit collaborative filtering. *Advances in Neural Information Processing Systems* 33 (2020), 1094–1105.

[12] Zhe Fu, Xi Niu, and Mary Lou Maher. 2023. Deep learning models for serendipity recommendations: a survey and new perspectives. *Comput. Surveys* 56, 1 (2023), 1–26.

[13] Chen Gao, Yu Zheng, Wenjie Wang, Fuli Feng, Xiangnan He, and Yong Li. 2024. Causal inference in recommender systems: A survey and future directions. *ACM Transactions on Information Systems* 42, 4 (2024), 1–32.

[14] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247* (2017).

[15] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[16] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).

[17] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[18] Juyong Jiang, Peiyan Zhang, Yingtao Luo, Chaozhuo Li, Jae Boum Kim, Kai Zhang, Senzhang Wang, Xing Xie, and Sunghun Kim. 2023. AdaMCT: adaptive mixture of CNN-transformer for sequential recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 976–986.

[19] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 197–206.

[20] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[21] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-interest network with dynamic routing for recommendation at Tmall. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 2615–2623.

[22] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 1419–1428.

[23] Jiacheng Li, Yujie Wang, and Julian McAuley. 2020. Time interval aware self-attention for sequential recommendation. In *Proceedings of the 13th international conference on web search and data mining*. 322–330.

[24] Ming Li, Naiyin Liu, Xiaofeng Pan, Yang Huang, Ningning Li, Yingmin Su, Chengjun Mao, and Bo Cao. 2023. FAN: Fatigue-Aware Network for Click-Through Rate Prediction in E-commerce Recommendation. In *Database Systems for Advanced Applications: 28th International Conference, DASFAA 2023, Tianjin, China, April 17–20, 2023, Proceedings, Part IV*. Springer, 502–514.

[25] Xuewei Li, Aitong Sun, Mankun Zhao, Jian Yu, Kun Zhu, Di Jin, Mei Yu, and Ruiguo Yu. 2023. Multi-Intention Oriented Contrastive Learning for Sequential Recommendation. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 411–419.

[26] Jianxun Lian, Iyad Batal, Zheng Liu, Akshay Soni, Eun Yong Kang, Yajun Wang, and Xing Xie. 2021. Multi-Interest-Aware User Modeling for Large-Scale Sequential Recommendations. *arXiv preprint arXiv:2102.09211* (2021).

[27] Guanyu Lin, Chen Gao, Yinfeng Li, Yu Zheng, Zhiheng Li, Depeng Jin, and Yong Li. 2022. Dual contrastive network for sequential recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 2686–2691.

[28] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130* (2017).

[29] Hao Ma, Xueqing Liu, and Zhihong Shen. 2016. User fatigue in online news recommendation. In *Proceedings of the 25th International Conference on World Wide Web*. 1363–1372.

[30] Daisuke Moriwaki, Komei Fujita, Shota Yasui, and Takahiro Hoshino. 2019. Fatigue-Aware Ad Creative Selection. *arXiv preprint arXiv:1908.08936* (2019).

[31] Qi Pi, Weijie Bian, Guorui Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Practice on long sequential user behavior modeling for click-through rate prediction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2671–2679.

[32] Yuhan Quan, Jingtao Ding, Chen Gao, Nian Li, Lingling Yi, Depeng Jin, and Yong Li. 2023. Alleviating Video-length Effect for Micro-video Recommendation. *ACM Transactions on Information Systems* 42, 2 (2023), 1–24.

[33] Yuhan Quan, Jingtao Ding, Chen Gao, Lingling Yi, Depeng Jin, and Yong Li. 2023. Robust preference-guided denoising for graph based social recommendation. In *Proceedings of the ACM Web Conference 2023*. 1097–1108.

[34] 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*. 1441–1450.

[35] 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*. 565–573.

[36] Yu Tian, Jianxin Chang, Yanan Niu, Yang Song, and Chenliang Li. 2022. When multi-level meets multi-interest: A multi-grained neural model for sequential recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1632–1641.

[37] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *Proceedings of the ADKDD'17*. 1–7.

[38] Ruobing Xie, Cheng Ling, Shaoliang Zhang, Feng Xia, and Leyu Lin. 2022. Multi-granularity Fatigue in Recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 4595–4599.

[39] Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Jiajie Xu, Victor S Sheng S. Sheng, Zhiming Cui, Xiaofang Zhou, and Hui Xiong. 2019. Recurrent convolutional neural network for sequential recommendation. In *The world wide web conference*. 3398–3404.

[40] An Yan, Shuo Cheng, Wang-Cheng Kang, Mengting Wan, and Julian McAuley. 2019. CosRec: 2D convolutional neural networks for sequential recommendation. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 2173–2176.

[41] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Jundong Li, and Zi Huang. 2022. Self-supervised learning for recommender systems: A survey. *arXiv preprint arXiv:2203.15876* (2022).

[42] Zeping Yu, Jianxun Lian, Ahmad Mahmoody, Gongshen Liu, and Xing Xie. 2019. Adaptive User Modeling with Long and Short-Term Preferences for Personalized Recommendation.. In *IJCAI*. 4213–4219.

[43] Yu Zheng, Chen Gao, Jianxin Chang, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2022. Disentangling long and short-term interests for recommendation. In *Proceedings of the ACM Web Conference 2022*. 2256–2267.

[44] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 5941–5948.

[45] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1059–1068.