Research Article
Multifeedback Behavior-Based Interest Modeling Network for Adaptive Click-Through Rate Prediction

Xiaoran Shang,1 Bangbo Huang,2 and Hongbin Ma3

1Liaoning Technical University, Fuxin, China
2Peking University, Beijing, China
3Beijing Institute of Technology, Beijing, China

Correspondence should be addressed to Hongbin Ma; mathmhb@bit.edu.cn

Received 26 May 2022; Revised 27 June 2022; Accepted 7 July 2022; Published 29 August 2022

With the rapid development of the Internet, the recommendation system is becoming more and more important in people’s life. Click-through rate prediction is a crucial task in the recommendation system, which directly determines the effect of the recommendation system. Recently, researchers have found that considering the user behavior sequence can greatly improve the accuracy of the click-through rate prediction model. However, the existing prediction models usually use the user click behavior sequence as the input of the model, which will make it difficult for the model to obtain a comprehensive user interest representation. In this paper, a unified multitype user behavior sequence modeling framework named as MBIN, a.k.a. multifeedback behavior-based Interest modeling network, is proposed to cope with uncertainties in the noisy data. The proposed adaptive model uses deep learning technology, obtains user interest representation through multihead attention, denoises user interest representation using the vector projection method, and fuses the user interests using adaptive dropout technology. First, an interest denoising layer is proposed in the MBIN, which can effectively mitigate the noise problem in user behavior sequences to obtain more accurate user interests. Second, an interest fusion layer is introduced so as to effectively model and fuse various types of interest representations of users to achieve personalized interest fusion. Then, we used auxiliary losses based on behavior sequences to enhance the effect of behavior sequence modeling and improve the effectiveness of user interest characterization. Finally, we conduct extensive experiments based on real-world and large-scale dataset to validate the effectiveness of our approach in CTR prediction tasks.

1. Introduction

With the gradual development of the Internet, users are often confronted with a huge amount of information. In order to reduce information overload and meet users’ diverse online service needs (e.g., e-commerce, short videos, and news), personalized recommendation systems have become increasingly important. After decades of development, industrial recommendation systems have now basically formed a two-stage funnel-type structure consisting of a recall phase and a ranking phase [1–4]. Click-through rate prediction, as the most central module in the ranking phase, usually directly determines the recommendation results and plays a significant role in user satisfaction and user dwell time [4, 5]. In addition, online advertising is the core product of major Internet companies, and the revenue of their advertisements is directly related to the accuracy of click-through rate estimation [1, 2]. Therefore, click-through rate prediction, as a common task in recommendation systems, has been widely studied in industry and also attracted the attention of many scholars in academia.

In recent years, inspired by the application of deep learning in fields such as computer vision and natural language processing, more and more work has introduced deep learning into click-through rate prediction models with remarkable results. At present, click-through rate prediction models based on deep learning can be broadly classified into three categories: (1) feature interactions—interactions...
between features are realized with the help of deep learning techniques [6–9], such as Wide&Deep [10] and AutoInt [11], and other models that use deep learning methods to extract feature interactions so as to improve the prediction accuracy of click-through rate prediction models; (2) user behavior sequence modeling—through RNN, the Transformer temporal modeling module is used to model user behavior sequences [1, 2, 4, 12], from which more accurate representations of user interests are extracted, and thus improve the effectiveness of the model. In this paper, we focus on user behavior sequence modeling, and the difficulty lies in how to extract more accurate user interests from real user behavior data and how to improve the model prediction based on user interest representations.

User behavior sequences usually reflect users’ interests [4, 12, 13], and the mining and utilization of user behavior sequences is usually the most important module in click-through rate prediction and even the whole recommendation system, which makes exploring how to model based on users’ historical behaviors a very important research direction in recommendation systems. In recent years, many works at home and abroad have attempted to model user behavior sequences [1, 2, 4], and some progress has been made. However, the existing works have only used historical user click behavior sequences for modeling [1, 2, 4], without considering other types of user behavior sequences. Recently, researchers have pointed out that user click behavior sequences alone are not sufficient to accurately extract user interest representations, and the behavior sequences of user disinterested feedback also represent user preferences to some extent [13, 14]. In most cases, user interaction data are with multiple feedback types, which generally include implicit positive feedback (e.g., click), implicit negative feedback (e.g., no click), and explicit positive feedback (e.g., like and buy), which are important for characterizing users’ unbiased interests [13, 14]. Therefore, how to model the behavior sequences of multiple feedback types of users is an important concern for the next-generation click-through rate prediction model based on user behavior sequence modeling.

In this study, the feedback data of users are divided into two categories: (1) implicit feedback—feedback behaviors that can reflect users’ interest/disinterest to some extent, with more but inaccurate data, including click and no-click feedback; (2) explicit feedback—feedback behaviors that can reflect users’ interest/disinterest very accurately, with less data, including rating, marking (like and dislikes), and other behaviors.

Implicit feedback usually contains a large amount of noise; e.g., the user may accidentally click on an item that is not of interest, or the user may not click on an item of interest because the slide is too fast. Therefore, how to denoise the data in implicit feedback is a key issue to improve the effectiveness of implicit feedback behaviors. Existing works on multitype behavior sequences [13, 14] only consider the modeling of multitype behavior sequences without considering the noise problem in user behavior sequences, and also do not consider the fusion problem of multitype behavior sequences, which will make the model unable to achieve the best results.

Based on the above observations, this paper proposes a new model that uses multitype user behavior feedback, thus improving the accuracy of the click-through rate prediction task. Specifically, in order to mitigate the impact of noisy behaviors in the user implicit feedback behavior sequences on the click-through rate model, this paper creatively introduces the user explicit behavior sequences into the click-through rate prediction model and uses the explicit feedback behavior sequences to denoise the implicit feedback behavior sequences, thus improving the effectiveness of user interest modeling. In addition, this paper designs a user interest interaction module to realize the adaptive fusion of user interest representation. The effectiveness of the proposed method is demonstrated by conducting sufficient experiments on a large-scale real dataset, and it is verified that the method can be extended to other models that require user behavior sequence modeling.

The main contributions of this work are presented as follows:

(i) A new framework for modeling user cancel-click behavior, MBIN, is proposed, which effectively combines the respective types of users and enriches the modeling of user interest representation

(ii) We propose an interest denoising layer that can effectively mitigate the noise problem in user behavior sequences and extract accurate user interests from implicit negative feedback

(iii) We propose an interest fusion layer that can effectively model and fuse various types of interest representations of users to achieve personalized interest fusion

(iv) We have conducted numerous experiments on real-world and large-scale datasets to validate the effectiveness of our approach in CTR prediction tasks

2. Click-Through Rate Prediction Model for User Behavior Sequence Modeling

The goal of click-through rate prediction models is to predict the real click probability of users on items, usually using deep neural network models based on binary classification. Specifically, such models input user attribute features (e.g., user ID, gender, and age), item attribute features (e.g., item ID and belonging category), and user behavior features (e.g., user behavior sequences) into the model after embedding layers, and then model them using structures such as multilayer perceptron (MLP) with nonlinear activation functions, and finally obtain a prediction score between 0 and 1, which is the predicted click-through rate of the item by the model.

3. User Behavior Modeling

The basic idea of user behavior sequence modeling is to use deep learning-based neural networks to model user historical behavior sequences in order to extract more accurate representations of user interests and thus improve the
accuracy of various model predictions, including click-through rate prediction models. Click-through rate prediction models based on user behavior sequence modeling have been favored by researchers related to industry as the most commonly used models in industry, and have also received attention from academia. Commonly used click-through rate prediction models based on user behavior sequence modeling include the deep interest network [1] (DIN), the deep interest evolution network [2] (DIEN), and the deep matching and ranking network [15] (DMR). All such models use user historical click-through behavior sequences for modeling. Most of these models add user historical behavior sequence features to the original click rate prediction model, and use sequence modeling modules such as recurrent neural network (RNN) and attention mechanism (Transformer’s model) to model user historical behavior, and input the user interest representation vector extracted by the model into the model after splicing and fusing with other feature vectors. The subsequent modules are used to obtain the final estimated click-through rate.

Usually, user behavior sequence modeling focuses on implicit positive feedback behaviors (clicks), while ignoring the rich but noisy implicit negative feedback behaviors (unclicks), precise but sparse explicit positive feedback behaviors (likes), and explicit negative feedback (dislikes), which are equally important for a comprehensive and accurate understanding of user interests. At present, there are very few research works in the area of considering users’ multiple feedback in click-through rate prediction models at home and abroad. For example, Xie et al. proposed a novel deep feedback network [16] (DFN) to capture users’ unbiased interest through their click, unclick, and dislike feedback. However, the existing methods have failed to consider the effect of noise in users’ implicit negative feedback on user interest extraction, which makes it difficult for existing methods to achieve the best results. Therefore, this paper proposes the multifeedback behavior-based interest modeling network (MBIN) framework, which can effectively model multiple types of user behavior sequences and improve model prediction by removing the noisy data from users’ implicit feedback through the user interest denoising module.

4. Methodology

We describe the technical details of MBIN in this section, which jointly utilizes multiple types of user historical behavior sequences for CTR estimation, including user implicit positive/negative feedback and explicit positive/negative feedback. Similar to the multifeedback in most other scenarios, we set click, unclick, and like behavior sequences as implicit positive, implicit negative, and explicit positive feedback, respectively. Figure 1 shows the overall architecture of MBIN. First, we present the interest denoising layer and the explicit interest contrastive layer, which are designed to handle the challenge of removing noise from implicit negative feedback (unclick) and obtaining adequate explicit positive feedback supervisory signals for long-tail items. Furthermore, we construct a user interest fusion layer to get user interests from behavior sequences and realize the adaptive fusion of various feedback embeddings. The details of each component are as follows.

4.1. Embedding Layer. The input of MBIN can be divided into three groups: user profile, target item, and three categories of user behavior sequences. The user profile is represented by \( P \), including corresponding features such as user id, age, and gender. We project the target item into \( I \), which refers to the candidate item for prediction and includes item id and brand id. In this paper, we model user multifeedback historical behavior sequences, which are lists of items and contain click sequence \( C = [C_1, C_2, \ldots , C_T] \), unclick sequence \( U = [U_1, U_2, \ldots , U_T] \), like sequence \( L = [L_1, L_2, \ldots , L_T] \), and dislike sequence \( D = [D_1, D_2, \ldots , D_T] \), where \( T \) is the maximum sequence length input to the model.

Since the features of the above model are categorical features, we use the embedding layer to transform all features to low-dimensional dense features. We denote \( e_p \) and \( e_t \) as the embedding vectors of the user profile and target item, respectively. Simultaneously, we project \( C, U, \) and \( L \) into the embedding space and obtain the output low-dimensional dense features \( e_c = [e_{c1}, e_{c2}, \ldots , e_{cT}] \), \( e_u = [e_{u1}, e_{u2}, \ldots , e_{uT}] \), \( e_l = [e_{l1}, e_{l2}, \ldots , e_{lT}] \), and \( e_d = [e_{d1}, e_{d2}, \ldots , e_{dT}] \). It is significant to mention that target item and three different behavior sequences use independent embedding layers, which usually achieve better performance [1].

4.2. Interest Extraction Layer. User historical behavior sequences can reflect users’ interest tendencies, and modeling the behavior sequences can effectively capture users’ interest representations, thus improving the accuracy of model prediction.

Specifically, the module mainly uses multihot target attention (MHTA) [3] for interest extraction of four different user behavior representations, using clicked behavior representations, unclicked behavior representations, liked behavior representations, and disliked behavior representations as inputs to MHTA, which can obtain clicked interest representations \( I_c \), unclicked interest representations \( I_u \), like interest representation \( I_l \), and dislike interest representation \( I_d \) respectively.

Concretely, we take the target item as query (Q), and \( e_c = [e_{c1}, e_{c2}, \ldots , e_{cT}] \), \( e_u = [e_{u1}, e_{u2}, \ldots , e_{uT}] \), \( e_l = [e_{l1}, e_{l2}, \ldots , e_{lT}] \), and \( e_d = [e_{d1}, e_{d2}, \ldots , e_{dT}] \) as key (K) and value (V) in their respective MHTA. The calculation of MHTA is shown in the following equation. The computation process is introduced with the clicked interest representation \( I_c \), as an example.

\[
\text{MHTA}(e_c, e_t) = \text{Concat}(\text{head}_1, \ldots , \text{head}_{H}) W^O, \\
\text{head}_h = \text{Attention}(e_c W^h, e_t W^h, e_u W^h, e_l W^h), \\
\text{Attention}(Q, K, V) = \text{softmax} \left( \frac{QK^T}{\sqrt{d_k}} \right) V, 
\]

where \( d_k \) is the key dimension.
where \( H \) is the number of headers, \( d_k \) is the dimension of the key vector \( k \) and query vector \( q \), and \( e_k = \{e, d_u\} \) is embedding of different behavior sequences. Here, \( W^O \) is a trainable linear matrix, and \( W^K_h, W^Q_h, \) and \( W^V_h \) are weighting matrices for the \( h \)-th head.

Similarly, the unclicked interest representation \( I_u \), the liked interest representation \( I_l \), and the disliked interest representation \( I_d \) can be obtained.

4.3. Interest Denoising Layer. The clicked interest representation \( I_c \) and the unclicked interest representation \( I_u \) in the implicit user feedback contain a lot of noise, and the purpose of the denoising layer is to denoise these two representations to obtain a cleaner vector of user interest representations.

In this paper, we believe that the noise in the click behavior sequence is caused by the user’s missclick behavior on the uninterested items, while the noise in the unclick behavior sequence is caused by the user’s failure to click on the interested items. Based on the above view, the implicit feedback can be denoised using the explicit feedback from users, i.e., denoising the unclicked interest behavior representation using the liked interest representation and denoising the clicked interest representation using the disliked interest representation. Therefore, two sets of orthogonal mapping pairs, \( \langle \text{unclick}, \text{like} \rangle \) and \( \langle \text{click}, \text{dislike} \rangle \), are constructed in this paper, and their corresponding combinations of sequence representations are \( \langle I_u, I_l \rangle \) and \( \langle I_c, I_d \rangle \), respectively.

Specifically, the mapped representation \( e_u^p \) of \( e_u \) in the direction of \( e_l \) is a mixed representation of the user’s unclicked preference and preferred preference, which is the noise of the sequence representation \( e_u \) of unclicked behavior, because this representation matches the user’s interest but appears in the unclicked behavior representation. In other words, by removing this postprojection representation, the denoised user unclicked behavior representation can be obtained. Similarly, the user’s click behavior after denoising can be obtained. The specific formula is as follows:

\[
I_u^p = \text{project}(I_u, I_l),
\]

\[
\text{project}(I_u, I_l) = \frac{I_u \cdot I_l}{||I_u|| \cdot ||I_l||}, \quad I_c
\]

\[
I_c^p = I_c - I_u^p.
\]

The purified representation \( I_u^p \) can be used to represent the pure unclicked interest preferences of users on the orthogonal space. Similarly, the purified click interest preference representation \( I_c^p \) can be obtained:

\[
I_c^p = \text{project}(I_c, I_d),
\]

\[
\text{project}(I_c, I_d) = \frac{I_c \cdot I_d}{||I_c|| \cdot ||I_d||}, \quad I_c
\]

\[
I_c^p = I_c - I_c^p.
\]
4.4. Interest Fusion Layer. After the interest extraction layer and the interest denoising layer, pure user click interest representation, non-click interest representation, like interest representation, and dislike interest representation can be obtained. Due to the dispersion of user behavior, some users may often mark items as like or dislike, while some users rarely mark them, and even some users’ click behavior is very sparse. Based on this observation, this paper creatively proposes an interest fusion layer based on dynamic dropout [17]. This module can capture the differences of user behavior distribution, effectively fuse user interests, and avoid overfitting the model to a certain kind of behavior representation. The core idea of dropout is to randomly discard a certain proportion of neurons in the process of model training, so as to reduce the risk of model overfitting. This paper believes that the huge difference in user behavior distribution may lead to the overfitting of the model to a certain kind of behavior representation, so an interest fusion layer based on dynamic dropout is designed. The model takes four real lengths as inputs of monotonic function to obtain four dropout ratios. The specific implementation of monotonic function is as follows:

$$p(S) = \left(1 + e^{\theta_1 - S \theta_2}\right)^{-1},$$

where \(s\) is the true length of the sequence, \(\theta_1\) and \(\theta_2\) are the hyperparameters controlling monotonicity and slope, and \(p(S)\) is the resulting dropout ratio.

Then, the four dropout ratios are taken as the real dropout probabilities of the four behavior sequences, so as to realize the adaptive fusion of four different types of behavior sequences in the training stage and slow down the impact of overfitting on the model. Specifically, four pure user interest representations are spliced after adaptive dropout to obtain the fused user interest representation \(I_f\). The calculation formula is as follows:

$$
\begin{align*}
\bar{I}_c &= \text{Concat}(I^c_f, p_c), \\
\bar{I}_u &= \text{Concat}(I^u_f, p_u), \\
\bar{I}_l &= \text{Concat}(I_l, p_l), \\
\bar{I}_d &= \text{Concat}(I_d, p_d).
\end{align*}
$$

4.5. Output Layer. After passing through the interest fusion layer, the unbiased fusion interest representation \(I\) of four user behavior sequences can be obtained \((I_f)\). Then, the fused user interest representation, user attribute representation, and item representation are spliced, and then output through sigmoid function after passing through multiple layers of full connection layers with activation function to obtain the user’s predicted click-through rate of items as shown in the following formula:

$$\bar{y} = \text{Sigmoid} \left(\text{MLP} \left(\text{Concat} \left(I_{\text{item}}, I_{\text{user}}, I_f\right)\right)\right),$$

where \(\bar{y}\) is the click-through rate predicted by the model.

4.6. Loss Function. When training the MBIN model, there are two optimization objectives: (1) When estimating the click-through rate of the target item, it is necessary to make the estimated value of the model close to the real value as much as possible; and (2) it is necessary to ensure that the interest representation obtained from the four behavior sequences of the model is as close as possible to the user’s real preferences of click, no click, like, and dislike. For objective (1), this paper optimizes by minimizing the cross-entropy loss function:

$$L_{\text{ctr}} = \frac{1}{N} \sum_{(x,y) \in [D]} -(y \log(p) + (1 - y) \log(1 - p)), \quad (7)$$

where \(D\) represents the training dataset of size \(N\), \(x\) represents the input sample of the MBIN model, \(y\) represents the predicted value of the model, and \(y \in \{0, 1\}\) represents whether the user has clicked on the item or not.

For objective (2), it is difficult to learn the desired results from the backpropagation point of view just by whether the user has clicked or not as a supervised signal. In order to enable the model to learn a representation that is more consistent with the real interest distribution of the user, this paper proposes an auxiliary loss-based method to help the user behavior sequence to be learned.

Specifically, for a sequence of real behaviors of length \(S\), MBIN uses the first \(S - 1\) user behaviors to predict the user’s \(S - \text{th}\) behavior, which is a mega multiclassification task, and the number of candidate items is the number of model classifications. Taking the user click behavior sequence as an example, the behavior sequence representations in the interest extraction layer after multihead target attention modeling are obtained, and the interest representations \(I_{S-1}^c\) represented by the first \(S - 1\) user behavior representations are averaged over the first \(S - 1\) user behavior representations to obtain the user’s first \(S - 1\) click behavior. The probability of the user’s next click on item \(s\) after this \(S - 1\) behavior can be defined by the softmax function as follows:

$$p_t = \frac{\exp^{t_{t-1}^c \psi}}{\sum_{k=1}^{K} \exp^{t_{t-1}^c \psi}}, \quad (8)$$

where \(\psi^c\) is the characterization of the \(S - \text{th}\) click item. Taking the cross-entropy as the loss function, there are the following losses:

$$L_{\text{aux}}^C = \frac{1}{N} \sum_{i=1}^{N} \sum_{s=1}^{K} -y_i^s \log(p_i^s), \quad (9)$$

where \(t_{t-1}^c \in \{0, 1\}\) represents the label of the \(S - \text{th}\) item in the sequence of user click behaviors, \(p_i^s\) is the corresponding prediction result, and \(K\) is the number of multiple classifications, i.e., the number of items. \(y_i^s = 1\) when and only when item \(s\) is the \(S - \text{th}\) behavior in the sequence of user click behaviors. Considering that the number of items is in the order of millions, which leads to too much computation for the softmax classification, the negative sampling technique [18] is used to simplify the computation, and the corresponding loss becomes as follows:
\[ L_{\text{NS}}^C = \frac{1}{N} \sum_{i=1}^{N} \log \left( \text{Sigmoid}\left( L_{i}^{C-1}\right) \right), \]

\[ + \sum_{h=1}^{H} \log \left( \text{Sigmoid}\left( -L_{h}^{C-1}\right) \right), \]

where \(\nu_h^C\) is the representation of negative samples obtained from negative sampling, and \(H\) is the number of negative samples sampled, whose value is much smaller than the total number of overall item \(C\).

Similarly, the auxiliary losses \(L_{\text{NS}}^U, L_{\text{NS}}^L,\) and \(L_{\text{NS}}^D\) for unclicked behavior, liked behavior, and disliked behavior are obtained.

Finally, the overall loss of the MBIN model is defined as follows:

\[ L_{\text{total}} = L_{\text{ctr}} + L_{\text{NS}}^C + L_{\text{NS}}^U + L_{\text{NS}}^L + L_{\text{NS}}^D. \]

### 5. Experiments

In order to verify the validity of the proposed model in this paper, sufficient experiments are conducted on a real large-scale dataset (Alimama) to evaluate the effectiveness of the MBIN model by comparing several state-of-the-art (SOTA) methods.

#### 5.1. Data Description.

In this paper, we conducted sufficient experiments based on the Alimama dataset and click ad data set, the details of which are listed in Table 1.

The Alimama dataset (https://tianchi.aliyun.com/dataset/dataDetail?dataId=56) consists of randomly selected ad display and click logs from Taobao over an 8-day period. It contains about 26 million logs, 1.15 million users, and 850,000 items. The logs of the first 7 days are used as training data and the logs of the last day are used as test data.

This dataset contains data on multiple types of user feedback behaviors, and includes purchases and favorites as explicit positive feedback (likes) and multiple shows but no clicks as explicit negative feedback (dislikes). Since the Alimama dataset is collected from real advertising scenarios with large-scale and complete features, it is widely used to study various CTR prediction models [1, 2, 12].

#### 5.2. Evaluation Metrics.

We evaluate the CTR prediction task in our dataset mentioned above. Specifically, the evaluation metric of our model is the area under the curve (AUC) of the receiver operating characteristic curve (ROC), which reflects the ranking ability of the model and is widely used in binary classification problems. It is worth noting that in real large-scale datasets, an AUC improvement of 0.1 is often a more substantial improvement, which usually leads to larger business gains. Furthermore, similar to DFN [16], we introduce RelaImpr to measure the improvement relative to the base model, which is calculated as follows:

\[ \text{RelaImpr} = \left[ \frac{\text{AUC}(\text{model}) - 0.5}{\text{AUC}(\text{baseline}) - 0.5} - 1 \right] \times 100\%. \]

### 5.3. Competitors and Experimental Settings

#### 5.3.1. Competitors.

In this paper, several state-of-the-art (SOTA) comparison schemes are implemented in the above real dataset, as follows:

1. **Wide&Deep.** It [10] consists of a wide part and a deep part, allowing the wide part to perform low-order feature interactions and the deep part to perform high-order feature interactions, thus enabling the model to learn both low-order and high-order feature interactions.

2. **AutoInt.** It [11] introduces a self-attentive neural network for feature interactions.

3. **DIN.** It [1] uses the target attention mechanism to model the relationship between candidate items and the user’s historical behavior sequence, which is widely used in behavior sequence modeling scenarios.

4. **DSIN.** It [12] divides the user’s historical click behavior sequence into multiple different sessions according to the temporal relationship of different click behaviors, and then uses Bi-LSTM to model the multiple sessions of the user to obtain the evolution of the user’s interest.

5. **DIEN.** It [2] utilizes a two-layer GRU to model user interest evolving process and extract user latent temporal interest.

6. **DMR.** It [15] combines the idea of collaborative filtering in matching methods for the ranking task in CTR prediction with user-to-item network and item-to-item network, respectively.

7. **DFN.** It [16] models the user’s explicit positive feedback, explicit negative feedback, implicit positive feedback, and implicit negative feedback behavior sequences to obtain an unbiased interest representation of the user.

8. **DMT.** In order to effectively model multiple types of user behavior sequences, DMT [14] used multiple different transformers to model the user’s click sequences, collection sequences, and purchase sequences, and extracted the interest representations of multiple user behaviors.

| Table 1: Detailed statistics of the Alimama dataset. |
|-----------------------------------------------|
| Content            | Alimama dataset |
| Num. of feedback records | 26,557,961 |
| Num. of users       | 1, 106, 768 |
| Num. of items       | 846, 811 |
| Num. of train data  | 23, 249, 296 |
| Num. of test data   | 3, 308, 665 |
| Num. of clicks      | 1, 366, 056 |
| Num. of unclicks    | 25, 191, 905 |
| Num. of likes       | 65, 903 |

All models are tested at least 10 times to filter out the optimal hyperparameters and report the best experimental results.
5.3.2. Parameter Settings. In this paper, all models mentioned in the paper were implemented using TensorFlow and experimentally evaluated using the exact same training and test datasets. The maximum length $T$ of each behavior sequence is set to 50. In the embedding layer, the embedding vector dimension is uniformly set to 64. The model batch data size is set to 512, and the learning rate is set to 0.0005. The hyperparameters of each method are heavily tuned, and the best hyperparameters are finally selected as the experimental results.

5.4. Overall Performances. As shown in Table 2, the following conclusions can be obtained: First, the proposed method in this paper achieves the best results, and achieves a relative improvement of 10.62% compared with Wide&Deep, which is a very significant improvement for real industrial-grade datasets. Second, by comparing the DIN, DMR, DFN, and DMT models, it can be obtained that using more types of user feedback behaviors can effectively improve the effectiveness of the models. Besides, both DFN and DMT models use different types of user behavior sequences, and their effectiveness is significantly higher than that of the DIN and DMR models, which only use user click behavior sequences. The experimental results are also the starting point of this paper for modeling multiple different user behaviors—richer user behaviors, which can help the models to obtain more accurate user interest representations.

5.5. Ablation Study

5.5.1. Effectiveness of the Interest Denoising Layer. The interest denoising layer is designed to purify the presentation of implicit negative feedback with noise. In order to explore the effectiveness of using like representation to denoise unclick representation by denoising module, we design three different denoising modules for comparison, namely, MBIN-IDN, which removes the interest denoising layer; MBIN_DUMN [19], which utilizes a vector project layer to get the vector representation after denoising; and MBIN_DDM [20], which gets the representation purified by the confidence layer and triplet loss.

As shown in Table 3, the effect is significantly reduced after removing the interest denoising layer, indicating that the interest denoising layer in the MBIN model can effectively remove the noise in the implicit user feedback and improve the prediction accuracy of the model.

5.5.2. Effectiveness of the User Interest Fusion Layer. In most scenarios, user interaction data have multiple feedback types, usually including implicit positive feedback (e.g., click), implicit negative feedback (e.g., no-click), explicit positive feedback (e.g., like and buy), and explicit negative feedback (e.g., multiple no-clicks), which are important for characterizing users’ unbiased interests. From a modeling perspective, how the four different types of user interest representations are fused will have a meaningful impact on the effectiveness of click-through prediction tasks.

5.5.3. Effectiveness of the Auxiliary Loss. In order to verify the effectiveness of the auxiliary loss function, the auxiliary loss function is removed from the MBIN model for the ablation experiment. In this paper, we define MBIN-L as the MBIN model with all behavior sequences of auxiliary loss functions removed to verify the effect of behavior sequences of auxiliary loss functions on the model effect. Similarly, MBIN-LC, MBIN-LU, MBIN-LL, and MBIN-LD are the models of the MBIN model after removing the auxiliary loss functions of clicked, unclicked, liked, and disliked behavior sequences, respectively. The experimental results of the above models are shown in Table 5.

The above experiments show that the effect of the model after removing the auxiliary loss functions in the four behavior sequences shows different degrees of decrease, among which the effect of removing the auxiliary loss functions in the click behavior sequences is the most obvious. Therefore, the auxiliary loss function designed in the MBIN model can effectively improve the ability of model interest modeling, and thus improve the model prediction accuracy.

5.6. Parameter Analysis. In order to analyze the effects of some important parameters in the MBIN model, parameter sensitivity experiments were conducted for $\theta_1$ and $\theta_2$ in the fusion layer of interest, and the experimental results are shown in Figures 2 and 3.
From the figures, it can be found that the overall trend of the model both increases and decreases as the values of $\theta_1$ and $\theta_2$ are taken. The hyperparameter $\theta_1$ determines the intercept of the monotonic function $p(S)$ and reaches the best result when it takes the value of 20. Intuitively, after choosing the optimal hyperparameters $\theta_1$ and $\theta_2$, the dropout ratio increases as the actual length of the user behavior sequence $S$ increases; i.e., the feature is more likely to be “dropped”; i.e., the risk of overfitting the model to the feature is reduced, thus making the model more robust.

6. Conclusions

In this paper, by introducing multitype behavior sequences of users into the click-through rate prediction model, the model is able to better capture user interests, thus improving the effectiveness of click-through rate prediction and ultimately the accuracy of the recommendation system. Then, by designing the interest denoising layer, the influence of noisy data in the implicit user feedback on the user interest extraction process is mitigated, thus further improving the effectiveness of the model. In addition, this paper designs an adaptive dropout-based user interest fusion layer, which can adapt to the variability of user behavior distribution and thus effectively fuse multiple types of user interest representations. After that, an auxiliary loss function based on behavior sequences is designed to help the model obtain more accurate interest representations. Finally, this paper demonstrates the rationality of the proposed method through sufficient experiments and verifies the effectiveness of each layer through ablation experiments. Our method improves AUC by 10.62% and can provide users with more accurate recommendation results.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no competing interests.

References

[1] G. Zhou, X. Zhu, C. Song et al., “Deep interest network for click-through rate prediction,” in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1059–1068, Association for Computing Machinery, London, U K, July 2018.

[2] G. Zhou, N. Mou, Y. Fan et al., “Deep interest evolution network for click-through rate prediction,” in *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence*, vol. 33, pp. 5941–5948, Hawaii, USA, January 2019.

[3] Q. Chen, C. Pei, S. Lv, C. Li, J. Ge, and W. Ou, “End-to-end user behavior retrieval in click-through rate prediction model,” 2021, https://arxiv.org/abs/2108.04468.

[4] Q. Chen, H. Zhao, W. Li, P. Huang, and W. Ou, “Behavior sequence transformer for e-commerce recommendation in alibaba,” in *Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data*, pp. 1–4, New York, NY, USA, August 2019.
[5] K. Zhang, H. Qian, Q. Cui et al., "Multi-interactive attention network for fine-grained feature learning in ctr prediction," in Proceedings of the 14th ACM International Conference on Web Search and Data Mining, pp. 984–992, New York, NY, USA, March 2021.

[6] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "Deepfm: a factorization-machine based neural network for ctr prediction," 2017, https://arxiv.org/abs/1703.04247.

[7] J. Lian, X. Zhou, F. Zhang, Z. Chen, X. Xie, and G. Sun, "Xdeepfm: combining explicit and implicit feature interactions for recommender systems," in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1754–1763, London, U K, July 2018.

[8] R. Wang, B. Fu, G. Fu, and M. Wang, "Deep & cross network for ad click predictions," in Proceedings of the ADKDD’17, pp. 1–7, Halifax, NS, Canada, August 2017.

[9] R. Wang, R. Shivanna, D. Cheng et al., “Dcn v2: Improved Deep & Cross Network and Practical Lessons for Web-Scale Learning to Rank Systems,” in Proceedings of the Web Conference 2021, pp. 1785–1797, Ljubljana, Slovenia, April 2021.

[10] H.-T. Cheng, L. Koc, J. Harmsen et al., "Wide & deep learning for recommender systems," in Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, pp. 7–10, Boston, MA, USA, September 2016.

[11] W. Song, C. Shi, Z. Xiao et al., "Autooint: automatic feature interaction learning via self-attentive neural networks," in Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1161–1170, Beijing, China, November 2019.

[12] Y. Feng, F. Lv, W. Shen et al., "Deep session interest network for click-through rate prediction," 2019, https://arxiv.org/abs/1905.06482?amp=1.

[13] C. Wu, F. Wu, T. Qi, and Y. Huang, "Feedrec: news feed recommendation with various user feedbacks," 2021, https://arxiv.org/abs/2102.04903.

[14] Y. Gu, Z. Ding, S. Wang, L. Zou, Y. Liu, and D. Yin, "Deep multifaceted transformers for multi-objective ranking in large-scale e-commerce recommender systems," in Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 2493–2500, Ireland, October 2020.

[15] Z. Lyu, Y. Dong, C. Huo, and W. Ren, "Deep match to rank model for personalized click-through rate prediction," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 156–163, 2020.

[16] R. Xie, C. Ling, Y. Wang, R. Wang, F. Xia, and L. Lin, "Deep feedback network for recommendation," in Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, pp. 2519–2525, Yokohama, Japan, January 2021.

[17] V. Tinto, “Dropout from higher education: a theoretical synthesis of recent research," Review of Educational Research, vol. 45, no. 1, pp. 89–125, 1975.

[18] Y. Goldberg and O. Levy, “word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method,” 2014, https://arxiv.org/abs/1402.3722.

[19] Z. Huang, M. Tao, and B. Zhang, "Deep user match network for click-through rate prediction," in Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1890–1894, Canada, July 2021.

[20] F. Lv, M. Li, T. Guo et al., "Xdm: improving sequential deep matching with unclicked user behaviors for recommender system," 2020, https://arxiv.org/abs/2010.12837.