IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 20 June 2023, accepted 13 September 2023, date of publication 28 September 2023, date of current version 11 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3320142

RESEARCH ARTICLE

Deep Adaptive Interest Network for CTR Prediction

JIANGUO WEI, LEI WANG, AND MEILING GE
School of Computer Engineering, Weifang University, Weifang 261061, China
Corresponding author: Meiling Ge (gemeiling@wfu.edu.cn)

ABSTRACT Click-Through Rate (CTR) prediction is important in many industrial applications, such as E-commerce, news, and information. Understanding sophisticated feature interactions behind users' behaviors is essential for CTR prediction. Although existing methods have made significant improvements, there exist some problems: (1) only concentrate on modeling implicit information from the user side, while ignoring the interest hidden in the historical interactive behaviors; (2) insufficient feature extraction, only focusing on high-order feature interactions or low-order feature interactions. To overcome these limitations, we propose a Deep Adaptive Interest Network (DAIN) for CTR Prediction in the local and global views, respectively. Specifically, to extract user's interest, we first develop a local attention mechanism applied to the user behaviors and candidate ads, which can adaptively calculate users' interest representation given a candidate ad in the local views. To capture feature interactions, we propose a feature interaction extractor containing Multi-layer Perceptrons (MLP) and Factorization Machines (FM) components to capture high-order and low-order feature interactions. To adaptively learn the influences of high-order and low-order feature interactions on the target item, we finally employ a linear-based global attention mechanism in the feature interaction extractor. The effectiveness of DAIN is verified by comprehensive experiments on three datasets.

INDEX TERMS Click-through rate prediction, recommendation system, machine learning, attention mechanism.

I. INTRODUCTION

With the development of Internet services and mobile devices [1], Internet users can conveniently access a large number of online products and services, such as online news [2], online shopping, e-education, and so on. However, While enjoying the convenience of the Internet, people are also facing the problem of information overload. To reduce information overload and meet users' needs, a recommendation system has been developed and plays an increasingly important role in modern life. It aims to help users to select the appropriate information [3] from the massive information (products, services) of Internet platforms and has been successfully applied to various online Internet platforms personalized recommendations. According to the report data [4], the recommendation system has brought huge sales revenue to some companies such as Amazon and Taobao. Therefore, establishing an effective recommendation system is significant in improving user experience and company revenue.

One of the key tasks in the recommendation system [5], [6] is to predict the click-through rate. In many recommendation systems, the target is to maximize the number of clicks, so that recommended items can be ranked based on estimated click-through rates [7], [8]. In addition, ad click-through rate prediction in online advertising systems is also essential to increase system revenue, because the ranking strategy of ads can be adjusted through click-through rates and bids. For example, in the advertising industry, a common way is that advertisers pay publishers only when the ads are clicked. So, no matter what happens, correctly estimating CTR is critical. As its importance, more and more researchers are working on click-through rate prediction.

At present, CTR prediction methods can be broadly divided into two kinds. One is the traditional CTR prediction, and the other is the CTR prediction based on deep learning.

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo.

Traditional CTR prediction [9], [10], such as the classical linear models [11], [12], are simple but exist limitations in learning feature interactions, such as: 1) lack of the ability of learning feature interactions and 2) over-reliance on manually extracted feature interactions. Therefore traditional methods are not feasible in large-scale systems. With the wide application of deep learning [13], [14] in many fields [15], [16], [17], Many CTR models have changed from traditional methods to deep CTR models. Deep learning models systematically extract higher-level and more abstract feature representations from the input data through a series of successive layers of nonlinear activation functions, which enable complex transformations and feature combinations. For example, Product based Neural Networks (PNN) [18], Deep Crossing Networks [19], Wide&Deep Models [20], and Deep Interest Networks (DIN), etc., these methods enhance the performance of the models by employing multi-layer non-linear neural networks to automatically extract high-order feature interactions [21], [22]. Similar models, Deep&Cross Networks [23], and Deep Neural Networks (DNN) [24] have improved CTR prediction to some extent. However, this type of approach has two limitations. First, these models only capture high-order feature interactions. According to the view of the Wide&Deep [20], considering both high-order and low-order feature interactions lead to additional improvements than considering either case alone. In other words, extracting low-order feature interactions is also very important for the recommendation. Second, these models do not take into account the user's representation of interest. So they lack a good explanation of which combinations of features make sense. However, the explanation of the CTR model allows advertisers to answer why users see specific advertisements and help users make wise decisions. In addition, the transparency and effectiveness of the prediction process should be established among users on the advertising platform. Therefore, we are looking for an approach that can extract the users' interests and capture both high-order and low-order feature interactions.

In this paper, we propose such an approach with a Deep Adaptive Interest Network(DAIN) motivated by the above limitations of existing methods. Specifically, the proposed method can automatically extract user's interests through historical behaviors and candidate ids. To capture feature interactions, we design a feature interaction extractor containing three parts: MLP, FM component, and linear-based global attention mechanism. Among them, MLP is to capture high-order feature interactions and FM component is to capture low-order feature interactions. Moreover, Considering high-order and low-order feature interactions play different roles in CTR prediction, from the global point of view, we propose a linear-based global attention mechanism to monitor. The main contributions of this paper are as follows:

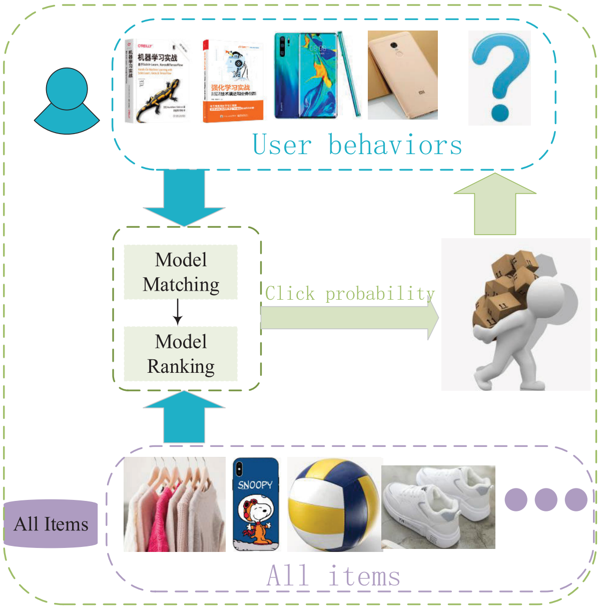

FIGURE 1. Description of running process of the advertising system. In the advertising system, related candidate ad lists are generated for users via methods such as collaborative filtering, and the top recommendations are selected according to the click-through rate to the user.

- To extract the user's interest hidden in the historical interactive behaviors and to monitor the importance of high-order and low-order feature interactions, we propose a hierarchical attention mechanism. This mechanism is used to mine the auxiliary information contained in the features (including composite features) of the user and item, and explore their different contributions to recommendation results from both local and global views. First, this mechanism adaptively calculates the user's interest representation according to historical behaviors and candidate ads from a local perspective. Second, this mechanism makes high-order and low-order feature interactions play different values. The proposed hierarchical attention mechanism increases the explanation of the model and substantially improves the efficiency of CTR prediction.
- Since high-order and low-order feature interactions can play important roles in CTR prediction, we propose a feature interaction extractor module. In this module, MLP captures high-order feature interactions and the FM component captures low-order feature interactions. In addition, this module does not require feature engineering. Thus, this module eases the problem of insufficient feature extraction.
- We conduct a large number of experiments on Amazon datasets of Electronics, Beauty, and Office_Products. Experimental results show that DAIN outperforms existing models.

II. RELATED WORK

A. CLICK-THROUGH RATE PREDICTION

CTR prediction is significant in recommendation systems and online advertising systems. Many Internet companies have proposed different models to improve CTR prediction. Shan et al. [19] propose a Deep Crossing method which automatically combines features to predict the click-through rate. However, the Deep Crossing model only considers high-order feature interactions. Cheng et al. [20] propose a Wide&Deep system that combines the linear and deep model to improve expression ability. Wang et al. [23] propose a Deep&Cross Network (DCN) to learn the high-order representation of features by applying a multi-layer residual structure. But DCN only considers high-order feature interactions, low-order feature interactions are important for CTR prediction either. At the same time, more and more researchers are paying attention to this issue in academics. For example, Zhang et al. [25] propose an FNN model that uses a deep neural network to learn valid modes automatically from feature interactions and predict users' ad clicks, but they ignore the low-order feature interactions. Qu et al. [18] propose the PNN model which introduces the product layer to capture high-order feature interactions. However, PNN captures little low-order feature interactions. These CTR prediction models improve the performance of CTR prediction to some extent.

B. LEARNING FEATURE INTERACTIONS

The key to CTR prediction is Learning feature interactions which is mainly divided into low-order and high-order feature interactions. For low-order feature interactions, FM [26] is a well-known example, which is proposed to capture the low-order feature interactions. It has been proved to be efficient for many tasks. Afterward, various variations of FM have been proposed. For example, Juan et al. [27] propose Field-aware Factorization Machines (FFM) to simulate the fine-grained interactions between features from different fields. Xiao et al. [28] propose Attentional Factorization Machines (AFM) which which takes into account the importance of interaction between different second-order features. However, these methods only consider low-order feature interactions. Recently, with the application of deep learning in CTR prediction, people have researched high-order feature interactions, such as PNN [18], Wide&Deep [20], and DIN [4] have achieved good performance in modeling high-order feature interactions using feed-forward neural networks. These CTR prediction models [29], [30] follow a similar model structure that combine the embedding layer and the MLP for learning the feature combination relation of different levels. This kind of CTR prediction model greatly reduces feature engineering. However, this kind of CTR prediction model ignores low-order feature interactions. Thus, our model follows this model structure to capture high-order feature interactions. In the mean time, we believe that low-order feature interactions are important in CTR prediction and employ FM components to capture low-order feature interactions.

TABLE 1. Symbols and descriptions.

Symbols	Definition
z_F	one-hot vector of *UserFeature*
z_H	one-hot vector of *UserBehavior*
z_I	one-hot vector of *Ad*
z_C	one-hot vector of *Context*
x	the input of the network
$\hat{y}$	the output of the network
y	the label of data
e_H^N	the feature vector of the $N-th$ historical behavior of user u
W_H^N	the weight matrix of e_H^N in the embedding layer
$r_u(I)$	the interest representation of user u with respect to I
$f(.)$	the prediction function
r_I	the feature vector of *Ad*
r_F	the feature vector of *UserFeature*
r_C	the feature vector of *Context*
r_U	the feature vector of $user'sinterest$
d	the dimension of the embedding layer

C. ATTENTION MECHANISMS

With the widespread application of attention mechanisms in many fields [31], [32], [33]. Recently, some researchers use the attention mechanism to improve CTR prediction performance [34]. For example, Xiao et al. [28] propose AFM, which adds an attention net based on FM to generate a weight of cross-features to distinguish the importance of crossover terms. Zhai et al. [35] propose DeepIntent, learning to assign attention points to different word locations according to the importance of intent. Zhou et al. [4] propose DIN, which introduces the attention mechanism for learning the relevant parts of historical behaviors. However, these models exit defects in feature extraction and they only consider high-order or low-order feature interactions. According to DIN from Alibaba, we introduce the local attention mechanism to adaptively calculate the user interest representation vector. In this paper, we also introduce the linear-based global attention mechanism to make low-order and high-order feature interactions play different roles in CTR prediction.

III. THE PROPOSED DAIN APPROACH

A. PROBLEM FORMULATION

In the advertising system [36], the ads recommended to users refer to relevant ads that users have previously browsed before. Figure.1 briefly illustrates the advertising system process, which consists of three steps: firstly, when users come into the advertising system, related candidate ad lists are generated for users via methods such as collaborative filtering. Secondly, the CTR of each ad in the candidate ad list is predicted. Finally, the top recommendations are selected according to the click-through rate of the user. Unlike many recommendation systems, users do not search directly and have no clear intention. Therefore, when building the CTR prediction model, learning feature interactions from different fields and paying attention to the user's interest representation are both important. The question we need to solve is how do we use the the features of users and ads for CTR prediction. Some important symbols and descriptions are listed in Table 1.

1) DEFINITION

In our CTR prediction model, we employ four types of features: *User Feature*, *User Behavior*, *Ad* and *Context*.

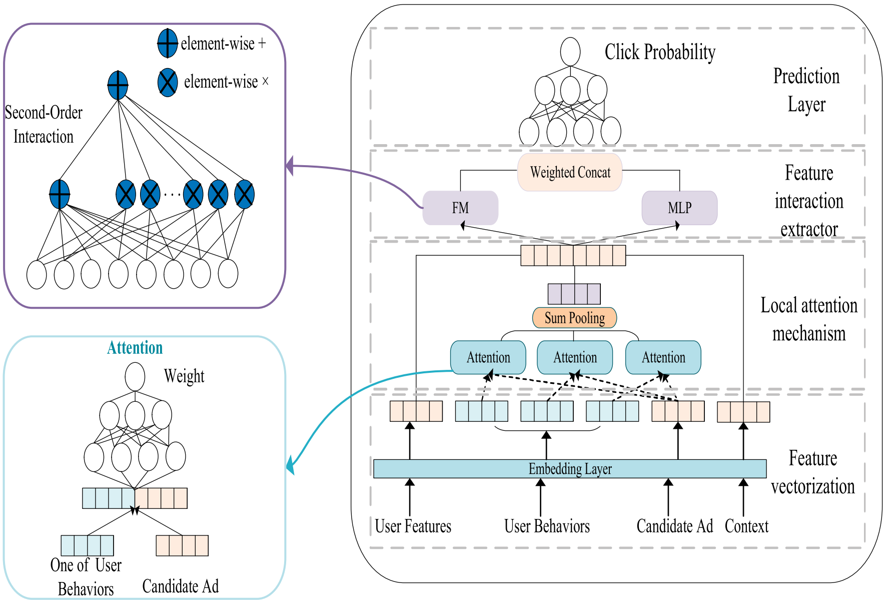

FIGURE 2. **The architecture of DAIN which contains feature vectorization, attention mechanism, feature interaction extractor, and Prediction layer. Feature vectorization module is to transform the original data into low-dimensional dense representations. Local attention mechanism is to extract interest. For Feature interaction extractor, MLP and FM capture high-order and low-order feature interactions, linear-based global attention mechanism is to monitor high-order feature interactions and low-order feature interactions. The prediction layer outputs the prediction result.**

In general, *User Feature's* fields are *gender*, *age*, and so on; *User Behavior's* fields contains the list of advertising ids that the user visits; The fields of *Ad* conclude *ad id*, *shop id*, and so on; *Context's* fields are made up of *type id*, *time*, and so on. The features of each field can be encoded into a one-hot [20] vector. Different feature's one-hot vectors from *User Feature*, *User Behavior*, *Ad*, and *Context* form z_F, z_H, z_I, z_C, respectively. In the sequential CTR model, it's clear that each field includes a list of behaviors, for example, each historical behavior of the user corresponds to an one-hot vector, which can be expressed as:

$$z_H = [H_1, H_2, H_3, \ldots, H_N] \in R^{S\times N} \tag{1}$$

where $H_N \in \{0, 1\}^S$ indicates N-*th* behavior, N represents the number of users' historical behaviors, and S represents the total count of ads that users can click.

2) PROBLEM

How do we utilize the features for CTR prediction? According to the above definitions, the problem can be formally turned into:

$$\begin{aligned} &Input : x\\ &Output : \hat{y}\\ &Constraints : \begin{cases} \hat{y} = f(x) \\ L = -\dfrac{1}{N}\displaystyle\sum_{(x,y)\in D}^{N} [y \log f(x) \\ \qquad +(1-y)\log(1-f(x))] \\ \arg\min(L) \end{cases} \end{aligned} \tag{2}$$

Therefore our goal is to find a model that takes feature combinations $x = \{z_F, z_H, z_I, z_C\}$ as input, under the constraints, it can output the click-through rate $\hat{y}$. $y \in \{0, 1\}$ as the label. $f(.)$ is the prediction function.

B. MODEL OVERVIEW

To settle the problem defined by Eq. (2), we propose a DAIN approach. Figure 2 shows the whole framework of DAIN, including a feature vectorization module for transforming the original data into low-dimensional dense representations, a local attention mechanism to extract interest, a feature interaction extractor to capture better feature extraction, and a prediction layer to output the prediction result. And we will introduce each module in detail.

1) FEATURE VECTORIZATION

Informative features play important roles in CTR prediction. It is essential to transform the original data into numerical data that can be processed by the neural network. Therefore, we employ one-hot [20] code to vectorize the original data. *User feature* z_F, *User Behavior* z_H, *Ad* z_I, *Context* z_C are inputted into the Feature vectorization layer, these features are transformed into numerical data and low-dimensional dense features. For example, for user historical behaviors $z_H = [H_1, H_2, H_3, \ldots, H_N] \in R^{S\times N}$, if the *i*-*th* ad is clicked in the N-*th* behavior, then H_N can be represented as:

$$H_N = [\underbrace{0\ldots0}_{i-1}10\ldots0] \tag{3}$$

where $H_N \in R^{1\times S}$ represents a one-hot vector with the dimension of S. Then, a mapping function is established to reduce high-dimensional binary vectors into low-dimensional dense representations. The mapping relationship in the feature vectorization layer is expressed as:

$$e_H^N = H_N W_H^N \tag{4}$$

where $W_H^N \in R^{S \times d}$ represents the corresponding weight matrix for H_N. d represents the feature dimension of the embedded vectors. e_H^N denotes the feature vector of the N-th value of e_H after the mapping in the feature vectorization layer. We obtain the feature vector of *User Behavior* $r_H = [e_H^1, e_H^2, e_H^3, \ldots, e_H^N]$. Similarly, according to Eq. (3), the one-hot vector of *User feature*, *Ad*, *Context* are z_F, z_I, z_C, respectively, and according to Eq. (4), z_F, z_I, z_C are all mapped to low-dimensional dense vectors r_F, r_I, r_C by feature vectorization layer respectively. After the above processing, we obtain the feature vector of *User feature* r_F, *User Behavior* r_H, *Ad* r_I and *Context* r_C.

2) LOCAL ATTENTION MECHANISM TO EXTRACT INTEREST

In most non-search advertising systems, users do not express their purchase intention directly. Designing models that capture users' interest from users' historical behaviors and candidate ad is important for improving CTR prediction. For example, a young girl visits the e-commerce website and sees the displayed mobile phone shell, and clicks it. It's obvious that the displayed ad hits the related interest of her historical behavior when she browses the mobile phone. However, different historical behaviors of the user will play different roles in expressing the user's interest. The attention mechanism originates from Neural Machine Translation (NMT) [31] which only pay attention on information relevant to the generation of the target word. In this layer, the local attention mechanism is proposed to learn about user interest representations after the feature vectorization layer. As is shown in Figure 2, it is applied to the user behaviors and candidate ads, which can adaptively calculate the user's interest representation $r_U(I)$:

$$\begin{aligned} r_U(I) &= g(r_I, e_H^1, e_H^2, \ldots, e_H^N) = \sum_{i=1}^{N} a(r_I, e_H^i) e_H^i \\ &= \sum_{i=1}^{N} a_i e_H^i \end{aligned} \tag{5}$$

where $g(.)$ represents local attention mechanism function, $\{e_H^1, e_H^2, \ldots e_H^N\}$ represents the embedding vectors of the user's historical behaviors of length N. a_i is the weight of each user's historical behavior. $a(.)$ is a MLP with one hidden layer and outputs the weight. It can be represented as:

$$a(r_I, e_H^i) = \sigma(relu([r_I, e_H^i] W_{at1}) W_{at2}) \tag{6}$$

where σ represents the *sigmoid* function [4], *sigmoid* and *relu* [20] are the activate functions, respectively. $W_{at1} \in R^{2d \times f_1}$ is the weight matrix of the activate unit *relu* and $W_{at2} \in R^{f_1 \times d}$ denotes the weight matrix of the activate unit *sigmoid*, respectively. f_1 is the number of neural units in the hidden layer. d represents the dimension of the embedded vectors. In this way, the different historical behaviors are weighted according to the candidate ad. And then the user's interest representation r_U can be extracted.

3) FEATURE INTERACTION EXTRACTOR

There is much valuable information behind implicit feature interactions for CTR prediction. The feature interaction extractor aims to extract feature interactions to better mine helpful information. This paper utilizes a three-layer fully connected network to capture high-order feature interactions. FM component is used to capture low-order feature interactions. And a linear-based global attention mechanism is to give different value to high-order and low-order feature interactions.

a: HIGH-ORDER FEATURE INTERACTIONS EXTRACTOR

High-order feature interactions are essential for CTR prediction model. In order to capture non-linear high-order feature interactions, this paper introduces a three-layer fully connected network, which contains an input layer, a hidden layer, and an output layer.

Formally, the definition of fully connected layers are as follows:

$$\begin{aligned} o_1 &= \sigma(tW_1 + b_1) \\ o_2 &= \sigma(o_1 W_2 + b_2) \\ o_3 &= o_2 W_3 + b_3 \end{aligned} \tag{7}$$

where $t = [r_F, r_U, r_I, r_C]$ represents the connection of the embedding vector of the user's feature, user's interest representation, candidate ad and content. $o_1 \in R^{1 \times l_1}$, $o_2 \in R^{1 \times l_2}$, $o_3 \in R^{1 \times l_3}$ represent the output of the input layer, hidden layer, and output layer, respectively. $W_1 \in R^{4d \times l_1}$, $W_2 \in R^{l_1 \times l_2}$, $W_3 \in R^{l_2 \times l_3}$ represent the weight matrix of the input layer, hidden layer, and output layer, respectively. $b_1 \in R^{1 \times l_1}$, $b_2 \in R^{1 \times l_2}$, $b_3 \in R^{1 \times l_3}$ are the vector of the bias. l_1, l_2, l_3 denote the number of neural units in the input layer, hidden layer, and output layer, respectively. This paper introduces the sigmoid as the activation function for each layer.

After the above processing, the high-order feature interactions o_3 are obtained.

b: LOW-ORDER FEATURE INTERACTIONS EXTRACTOR

Extracting Low-order feature interactions is also important for CTR prediction. The FM component [26] is a factorization machine used in the collaborative recommendation. It can not only capture linear feature interactions among features but also model pairwise feature interactions. In this paper, FM is to learn low-order feature interactions, sharing the same input with the MLP. The output of FM can be expressed as:

$$y_{FM} = \sum_{i=1}^{4} W_i t_i + \sum_{i=1}^{4} \sum_{j=i+1}^{4} W_{ij} t_i t_j \tag{8}$$

where $W_i \in R^4$ is the weight matrix reflects the importance of linear features, $t_1 = r_F, t_2 = r_U, t_3 = r_I, t_4 = r_C$. W_{ij} is the weight matrix reflecting the importance of pairwise features which is factorized as: $W_{ij} = v_i^T v_j$, where $v_i \in R^d$, $v_i \in R^d$ represents the embedding vector of feature i, and d is the size of the embedding vector.

c: LINEAR-BASED GLOBAL ATTENTION MECHANISM

This paper learns both high-order and low-order feature interactions by MLP and FM component. Considering high-order and low-order feature interactions might play different roles in CTR prediction, from the global point of view, this paper proposes a linear-based global attention mechanism to monitor. The linear-based global attention mechanism is a linear module, which is simple and can save computation time cost. Meanwhile, it achieves good results. The output of the linear-based global attention mechanism can be represented as:

$$v = (w_{FM} y_{FM} + w_{MLP} o_3) \tag{9}$$

where $w_{MLP} \in R$, $w_{FM} \in R$ represent the weight of the high-order and low-order feature interactions respectively.

4) PREDICTION LAYER

After the linear-based global attention mechanism is the prediction layer. The output of the prediction layer is as follows:

$$\hat{y} = \sigma(v) \tag{10}$$

where $\hat{y} \in \{0, 1\}$.

Ultimately, DAIN outputs the result of the prediction. The objective function is a negative log-likelihood function [37], denoted as:

$$L = -\frac{1}{N} \sum_{(x,y) \in D}^{N} [y \log \hat{y} + (1 - y) \log(1 - \hat{y})] \tag{11}$$

where D denotes the training set of size N and $y \in \{0, 1\}$ represents whether the user clicks the target item. $f(x)$ denotes the output of the network $\hat{y}$. The learning algorithm of DAIN is presented in Algorithm1. The main function of DAIN algorithm is to input feature combinations $x = \{z_F, z_H, z_I, z_C\}$, and then output the click-through rate $\hat{y}$.

IV. EXPERIMENTS

In this section, we describe our experiment settings and results in detail. We compare our proposed model with other advanced models on the Amazon dataset.

1) EXPERIMENT SETUP

The experiments are achieved on the framework of TensorFlow 2.10.0 with Python v3.9. We adopt the NVIDIA GTX 3090 TiGPU as the hardware environment.

Algorithm 1 AMLP-FM Algorithm

```
Require: User feature $z_F$, User UserBehavior $z_H$, Ad $z_I$,
    Context $z_C$;
    Parameters: $S$, $lr$, $N$, $L$ (the value of negative log-likelihood function);
Ensure: Initialize: $lr$=0.1, Random $w_i$, $b_i$;
repeat
    for number of training iteration do
        Feature vectorization for $z_F, z_H, z_I, z_C$ to get $r_F, r_H, r_I, r_C$;
        Calculate user's interest representation $r_U(I)$ according to Eq. (5)and Eq. (6);
        Capture High-order feature interactions according to Eq. (7);
        Capture low-order feature interactions according to Eq. (8);
        the linear-based global attention mechanism output the final feature according to Eq. (9);
        Prediction value $\hat{y}$ by putting $v$ and $w$ to Eq. (10);
        Calculate value $l$ of negative log-likelihood function based on $\hat{y}$ and $y$;
    end for
until The rate of change of L tends to be stable
```

TABLE 2. Statistics of datasets used in the paper.

Dataset	Users	Goods	Categories	Samples
Amazon(Elect)	192403	63001	801	1689188
Amazon(Beauty)	22363	12101	226	198502
Amazon(Office_Products)	4905	2420	279	53258

2) DATASETS

Amazon Dataset: Amazon dataset is the baseline dataset for CTR predictions [38], [39], [40], [41]. We use three subsets of the Amazon dataset: Electronics, Beauty, and Office_Products. Among them, the dataset of Electronics includes 192,403 users, 63,001 items, 801 categories, and 1,689,188 click-through behavior records, the dataset of Beauty includes 22,363 users, 12,101 items, 226 categories, and 198,502 samples, the dataset of Office_Products includes 4905 users, 2420 items, 279 categories, and 53258 samples. Table 2 shows the statistics of all the datasets. Features include items_id, cate_id, user reviewed items_id_list and cate_id_list. In the data sets, each user or item has more than 5 clicks. Let all behaviors of a user be $(H_1, H_2, \ldots, H_K, \ldots, H_N)$, the task is to predict the $(k + 1)$ *-th* reviewed items.

3) PARAMETER SETTINGS

We use repeated tests to determine the optimal super parameter of each algorithm. We apply Stochastic Gradient Descent (SGD) as an optimizer in all models. We also establish an automatic decay mechanism for learning rates. By iterative training, the learning rate gradually decreases from 1 and the decay rate is set to 0.1. The batch size is 32.

The embedded size of the model is 128, which is consistent with other comparison methods. The number of layers of the MLP neural network is 3, and the number of units per layer is set to 80, 40, 1.

A. EVALUATION METRICS AND BASELINES

1) EVALUATION METRICS

We employ two evaluation metrics, AUC (The Area Under the ROC Curve) and RelaImpr (Relative Importance) to evaluate the performance of different models. AUC is the total area under the ROC curve [42]. It measures the order of the ads by sorting the predicted click-through rates of all ads. We defined it as:

$$AUC = \frac{1}{m+m^-} \sum_{x^+ \in D^+} \sum_{x^- \in D^-} (g(f(x^+) > f(x^-))) \qquad (12)$$

where D^+ represents the set of all positive examples, D^- represents the set of all negative examples, respectively. $m+$ and $m-$ represent the numbers of positive samples and negative samples, respectively. $f(.)$ is the result of the model's prediction and $g(.)$ is the indicator function. Besides, we employ the RelaImpr metric [7, 21] to measure the relative improvement over models. For random guesses, AUC is 0.5. Therefore, RelaImpr is defined as:

$$RelaImpr = (\frac{AUC_{model1} - 0.5}{AUC_{model2} - 0.5} - 1) \times 100\% \qquad (13)$$

2) BASELINES

We compare the proposed model with the existing ten CTR prediction models, as follows:

LR [11]: Logistic regression (LR) is a weak baseline on the TensorFlow framework.

BaseModel [4]: Basemodel follows the embedding&MLP architecture for predicting click-through rates.

PNN [18]: PNN can be regarded as an improved version of Basemodel. After the embedded layer, a production layer is introduced to capture high-order feature interactions.

Wide&Deep [20]: Wide&Deep model concludes two parts, one is the wide part, handling the cross-product of manual design, and the other is the deep part, automatically extracting the nonlinear relationship of features.

DIN [4]: DIN uses the attention mechanism combined with the DNN algorithm to delicately activate related user behaviors and obtains the adaptive representative vector that changes in different advertisements.

GIN [43]: GIN introduces the diagram learning into the user's intention to dig at the user's intention, and proposes the end-to-end joint training of the sponsorship search and CTR prediction tasks.

DIEN [41]: DIEN learns user interest by learning sequence-based dependencies based on GRU, and proposes AUGRU to learn the trend of changes in user's interest.

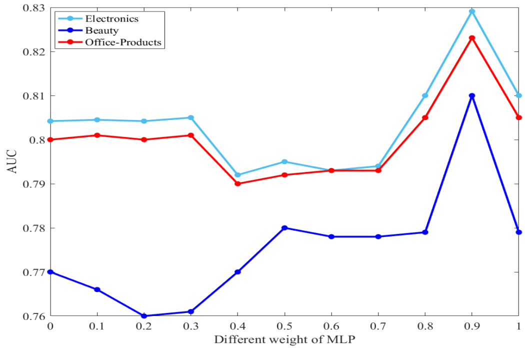

FIGURE 3. **Performance of different weight of high-order and low-order feature interactions.**

MIMN [44]: MIMN is a multi-channel memory network that processes user interest modeling with an infinite length of sequential behavior data.

DMIN [45]: DMIN models users' latent multiple interests for click-through rate prediction tasks.

TGIN [46]: TGIN introduces triangles in the neighborhood of the commodity-commodity diagram and views these triangles as basic units of user interest.

B. PERFORMANCE EVALUATION

1) PERFORMANCE OF DIFFERENT WEIGHT IN LINEAR-BASED GLOBAL ATTENTION MECHANISM

We conduct multiple experiments to explore the effect of the different weights of w_{FM} and w_{MLP} in a linear-based global attention mechanism which makes the high-order and low-order feature interactions play different values in CTR prediction. In the experiment, we make the sum of w_{FM} and w_{MLP} equal to 1, for example, when w_{MLP}=1, w_{FM}=0, and when w_{MLP}=0.5,w_{FM}=0.5. So in Figure 3, we just show the value of w_{MLP} for convenience. As shown in Figure 3, w_{MLP} goes from 0 to 1, so w_{FM} goes from 1 to 0. The DAIN model equals DIN when w_{MLP} is 1. We conduct five experiments on each value of w_{FM} for DAIN and obtain the average value of AUCs. As shown in Figure 3, whenw_{MLP} equals to 1, it represents we only capture high-order feature interactions, and when w_{MLP} equals to 0, it represents we only capture low-order feature interactions. When w_{MLP} equals to 0 and 1, we can conclude that compared with low-order feature interactions, high-order feature interactions play a higher role in predicting click-through rate. When w_{MLP} equals to 0.9 and w_{FM} equals to 0.1, AUC takes the maximum value on both datasets, this is because when the weight of the high-order feature is 0.9 and the weight of the low-order feature is 0.1, the fused feature is more consistent with the real feature. At the same time, when w_{MLP} equals to 0.8 and w_{MLP} equals to 1.0, they have the same result, because the distance between the features and the real features is consistent. We can draw a conclusion from the experimental result that in the small range of the best feature, the change of the feature has a large impact on the result, but in the large range, the change of the influence is small, such as the value of w_{MLP} goes from 0.1 to 0.7.

TABLE 3. Comparison on prediction performance. The results of these methods annotated by the '†' symbol are directly cited from the original papers or reference. And All the lines calculate RelaImpr by comparing with BaseModel on each dataset respectively.

Model	Amazon(Electronics)		Amazon(Beauty)		Amazon(Office_Products)	
	AUC	RelaImpr	AUC	RelaImpr	AUC	RelaImpr
LR	0.7102	-8.24%	0.7045	-6.8%	0.76207	-16.24%
BaseModel	0.7291	0	0.7195	0	0.8129	0
PNN	0.7543	10.9%	0.7379	8.38%	0.8132	0.09%
Wide&Deep	0.7456	7.2%	0.7378	8.33%	0.8130	0.03%
DIN†	0.7603	13,6%	0.7548	16.08%	0.8133	0.12%
GIN†	0.7684	17.1%	-	-	-	-
MIMN†	0.7563	11.8%	-	-	-	-
DIEN†	0.7792	21.8%	-	-	-	-
DMIN†	0.7778	21.2%	-	-	-	-
TGIN†	0.7944	28.5%	-	-	-	-
DAIN	0.8291	43.64%	0.8281	49.4 %	0.8231	3.25%

TABLE 4. Effect of FM component and hierarchical attention, all the other lines calculate RelaImpr by comparing with BaseModel.

Model	Amazon(Electronics)		Amazon(Beauty)		Amazon(Office_Products)	
	AUC	RelaImpr	AUC	RelaImpr	AUC	RelaImpr
BaseModel	0.7291	0	0.7195	0	0.8129	0
BaseModel + FM	0.760	13.48%	0.74	19.3%	0.8132	0.09%
BaseModel + attention mechanism	0.7603	13.61%	0.7548	16.08%	0.8133	0.127%
DAIN	0.8291	43.64%	0.8281	49.47%	0.8231	3.25%

2) THE EFFECT ON RECOMMENDATION DIVERSITY

This paper considers the application of recommendation system in different scenarios. For Electronics dataset, Beauty dataset, and Office_Products dataset, we use all the data in the dataset. All experiments are repeated 5 times and report the averaged results. For DIN, GIN, MIMN, DIEN, DMIN, and TGIN, we reproduce its code and obtain similar results to the published paper. We have the following observations: Table 3 illustrates all deep networks are significantly better than LR, which shows the power of deep learning. We observe that on the Electronics dataset, DAIN model outperforms LR, BaseModel, Wide&Deep model, PNN, DIN, GIN, MIMN, DIEN, DMIN, and TGIN by 11.89%, 10%, 7.48%, 8.35%, 6.88%, 6.07%, 7.28%, 4.99%, 5.13%, 3.47%, respectively, on AUC, and by 56.56%, 43.64%, 29.41%, 33.99%, 26.43%, 22.61%, 28.4%, 17.87%, 18.46%, 11.78% on RelaImpr, respectively. And we observe that on the Beauty dataset, the proposed DAIN outperforms LR, BaseModel, Wide&Deep model, PNN, and DIN by 6.10%, 1.02%, 0.99%, 1.01%, 0.98%, respectively, on AUC, and by 60.44%, 49.47%, 37.91%, 37.97%, 28.76% on RelaImpr, respectively. At the same time, we observe that on the Office_Products dataset, the proposed DAIN model outperforms LR, BaseModel, Wide&Deep model, PNN, and DIN by 12.36%, 10.86%, 9.02%, 9.03%, 7.33%, respectively, on AUC, and by 23.28%, 3.25%, 3.16%, 3.22%, 3.12% on RelaImpr, respectively. The AUC of PNN and Wide&Deep model are approximate. This is because their network structures are adjusted slightly based on BaseModel. Results show that a good network structure can improve the CTR prediction performance of the traditional DNN model. GIN is better than DIN, indicating that the introduction of map learning in CTR prediction can use user intentions to mine relieving the sparseness of behavior. DIEN uses a specially designed AUGRU to better simulate the evolution of interest. DMIN captures multiple interests to get good results. It is worth noting that our method can be better than all baselines on Electronics, Beauty and Office_Products datasets. This is because we not only propose a hierarchical attention mechanism to calculate user's interest and make feature interactions play different values but also we extract both high-order and low-order feature interactions for CTR prediction. Experimental results illustrate that our model is superior to the most advanced method on the CTR prediction task.

For Electronics dataset, we extract test data from two different groups: cold start user group and heavy user group. The data of the cold start user group is historical behavior 1 to 5; the heavy user group is users' data of the historical behavior of more than 23, we use LR, Basemodel, PNN, Wide&Deep and DIN as the comparison model. Figure 4 shows the results of cold start user group. The experimental results show that with the decrease in user historical behavior data, compared with the results on the overall data set of Table 3, the accuracy of all methods has decreased accordingly. And Wide&Deep recommendation accuracy decreases significantly. However, the DAIN method holds the best recommendation performance. It is observed that the model we proposed can better deal with the problem of cold start, because we use Stochastic Gradient Descent (SGD) as an optimizer and we also establish an automatic decay mechanism for the learning rate to deal with the overfitting problem. Meanwhile, our model employs a hierarchical attention mechanism that can not only calculate user's interest and make feature interactions

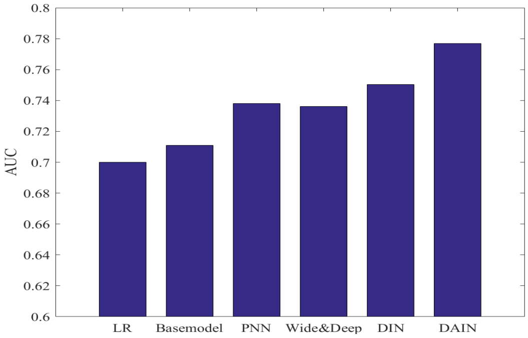

FIGURE 4. Cold start user group of experiment results.

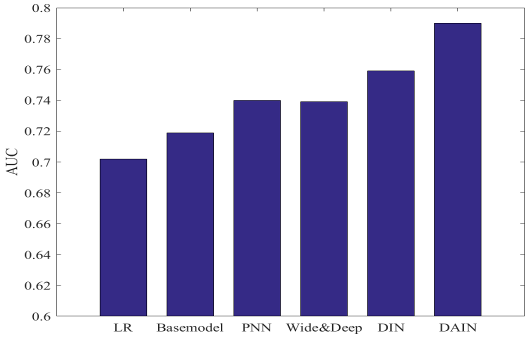

FIGURE 5. Heavy user group of experiment results.

make full use of CTR prediction. The hierarchical attention mechanism improves the performance of CTR prediction. The comparison experimental results of the heavy user group are shown in Figure 5. The results demonstrate that as the amount of data decreases, compared with the results on the overall data set in Table 3, the accuracy of all methods is reduced accordingly. However, the DAIN method maintains the best recommendation performance. The results show that all deep networks are significantly better than LR. The AUC of BaseModel, PNN, and Wide&Deep model is approximate. This is because their network structures are similar. And it is obvious that DAIN obtains good improvement. The excellence of DAIN may be related to the hierarchical attention mechanism.

3) PERFORMANCE OF ACTIVATION FUNCTION

We compare the performance of deep models when applying *sigmoid*, *relu*, and *tanh* on the dataset of Beauty. As shown in Figure 6, *sigmoid* is more appropriate than tanh and relu for all the deep models. Hence, we apply sigmoid in our paper.

C. APPLICATION STUDY

We will show the effect of the FM component and hierarchical attention model in this section.

1) EFFECT OF FM COMPONENT

The results of different CTR prediction methods is shown in Table 4. Compared to BaseModel, BaseModel + FM obtains obvious improvement. Compared to BaseModel, it's not different to find that not only do the high-order feature interactions captured by BaseModel play a certain role in the CTR prediction, but also the low-order feature interactions are important for CTR prediction.

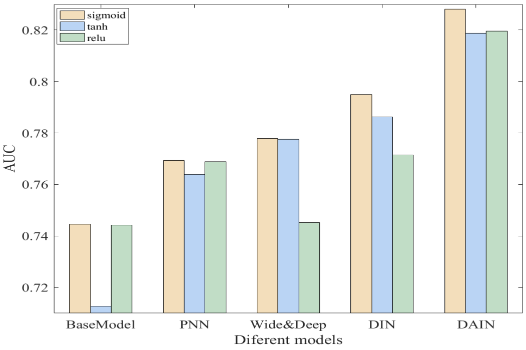

FIGURE 6. Activation function.

2) EFFECT OF HIERARCHICAL ATTENTION MODEL

Based on the obtained with BaseModel + FM, we further explore the effect of the hierarchical attention model on the datasets of Electronics, Beauty and Office_Products. As shown in Table 4, DAIN outperforms BaseModel, BaseModel + FM, and BaseModel + local attention mechanism respectively on three datasets. So we find the hierarchical attention model can bring great improvements. It is mainly by the following two aspects: First, we adaptively calculate the user's interest representation according to the candidate ad by using the local attention mechanism model, which can not only improve the efficiency and the accuracy of CTR prediction and increase the explanation of the model. Second, considering high-order feature interactions captured by MLP and low-order feature interactions captured by FM components play different roles in CTR prediction, we employ a linear-based global attention mechanism to distinguish their different importance. Consequently, our proposed method has achieved significant performance improvements.

V. CONCLUSION AND FURTHER WORK

In this paper, we propose a Deep Adaptive Interest Network (DAIN) that both learn high-order and low-order feature interactions in order to overcome the shortcomings of existing models and achieve better performance. It mainly includes the following advantages: 1. It learns both high-order and low-order feature interactions without feature engineering; 2. We propose a linear-based global attention mechanism to monitor the high-order and low-order feature interactions; 3. It explores users' interests based on their historical behaviors to predict CTR more efficiently. We conduct extensive experiments on the Amazon datasets of Electronics, Beauty and Office_Products to compare our model to the most advanced CTR prediction model. Experimental results testify that DAIN outperforms most advanced methods in the aspect of AUC and RelaImpr. In the future, we will study two directions to improve our model. One is introducing transformer to strengthen the ability of CTR prediction. The other is Considering long and short -term interests for recommendation.

REFERENCES

[1] G. Ran, J. Li, Q. Zhang, and X. Niu, "The association between social anxiety and mobile phone addiction: A three-level meta-analysis," *Comput. Hum. Behav.*, vol. 130, May 2022, Art. no. 107198.

[2] C. Wu, F. Wu, T. Qi, Q. Liu, X. Tian, J. Li, W. He, Y. Huang, and X. Xie, "FeedRec: News feed recommendation with various user feedbacks," in *Proc. ACM Web Conf.*, 2022, pp. 2088–2097.

[3] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 4, pp. 577–590, Jul. 2016.

[4] G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, and K. Gai, "Deep interest network for click-through rate prediction," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 1059–1068.

[5] S. Pandya, M. A. Wakchaure, R. Shankar, and J. R. Annam, "Analysis of NOMA-OFDM 5G wireless system using deep neural network," *J. Defense Model. Simul., Appl., Methodol., Technol.*, vol. 19, no. 4, pp. 799–806, Oct. 2022.

[6] J. Qiao, L. Wang, and L. Duan, "Sequence and graph structure co-awareness via gating mechanism and self-attention for session-based recommendation," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 9, pp. 2591–2605, Sep. 2021.

[7] Y. Feng, F. Lv, W. Shen, M. Wang, F. Sun, Y. Zhu, and K. Yang, "Deep session interest network for click-through rate prediction," 2019, *arXiv:1905.06482*.

[8] W. Song, C. Shi, Z. Xiao, Z. Duan, Y. Xu, M. Zhang, and J. Tang, "AutoInt: Automatic feature interaction learning via self-attentive neural networks," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 1161–1170.

[9] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, I. Bosnic, and E. Duval, "Context-aware recommender systems for learning: A survey and future challenges," *IEEE Trans. Learn. Technol.*, vol. 5, no. 4, pp. 318–335, Oct. 2012.

[10] M. Hassan and M. Hamada, "Smart media-based context-aware recommender systems for learning: A conceptual framework," in *Proc. 16th Int. Conf. Inf. Technol. Based Higher Educ. Training (ITHET)*, Jul. 2017, pp. 1–4.

[11] O. Chapelle, E. Manavoglu, and R. Rosales, "Simple and scalable response prediction for display advertising," *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 4, pp. 1–34, Jan. 2015.

[12] H. B. McMahan, G. Holt, D. Sculley, M. Young, D. Ebner, J. Grady, L. Nie, T. Phillips, E. Davydov, and D. Golovin, "Ad click prediction: A view from the trenches," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 1222–1230.

[13] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 2, pp. 766–775, Apr. 2020.

[14] Y. Zhang, X. Shi, H. Zhang, Y. Cao, and V. Terzija, "Review on deep learning applications in frequency analysis and control of modern power system," *Int. J. Elect. Power Energy Syst.*, vol. 136, 2022, Art. no. 107744.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[16] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6645–6649.

[17] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 1–9.

[18] Y. Qu, H. Cai, K. Ren, W. Zhang, Y. Yu, Y. Wen, and J. Wang, "Product-based neural networks for user response prediction," in *Proc. IEEE 16th Int. Conf. Data Mining (ICDM)*, Jun. 2016, pp. 1149–1154.

[19] Y. Shan, T. R. Hoens, J. Jiao, H. Wang, D. Yu, and J. Mao, "Deep crossing: Web-scale modeling without manually crafted combinatorial features," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 255–262.

[20] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, and M. Ispir, "Wide & deep learning for recommender systems," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, 2016, pp. 7–10.

[21] H. Wang, F. Zhang, X. Xie, and M. Guo, "DKN: Deep knowledge-aware network for news recommendation," in *Proc. World Wide Web Conf.*, 2018, pp. 1835–1844.

[22] H. Wang, F. Zhang, J. Wang, M. Zhao, W. Li, X. Xie, and M. Guo, "RippleNet: Propagating user preferences on the knowledge graph for recommender systems," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, 2018, pp. 417–426.

[23] R. Wang, B. Fu, G. Fu, and M. Wang, "Deep & cross network for ad click predictions," in *Proc. ADKDD*, 2017, pp. 1–7.

[24] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for YouTube recommendations," in *Proc. 10th ACM Conf. Recommender Syst.*, 2016, pp. 191–198.

[25] W. Zhang, T. Du, and J. Wang, "Deep learning over multi-field categorical data: A case study on user response prediction," in *Advances in Information Retrieval*. Padua, Italy: Springer, Mar. 2016, pp. 45–57.

[26] S. Rendle, "Factorization machines," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2010, pp. 995–1000.

[27] Y. Juan, Y. Zhuang, W.-S. Chin, and C.-J. Lin, "Field-aware factorization machines for CTR prediction," in *Proc. 10th ACM Conf. Recommender Syst.*, 2016, pp. 43–50.

[28] J. Xiao, H. Ye, X. He, H. Zhang, F. Wu, and T.-S. Chua, "Attentional factorization machines: Learning the weight of feature interactions via attention networks," 2017, *arXiv:1708.04617*.

[29] J. Zhang, J. Wang, H. Zhang, M. Miao, Z. Cai, and F. Chen, "Multi-level cross-attention Siamese network for visual object tracking," *KSII Trans. Internet Inf. Syst.*, vol. 16, no. 12, pp. 3976–3990, 2022.

[30] S.-J. Han and S.-B. Cho, "Evolutionary neural networks for anomaly detection based on the behavior of a program," *IEEE Trans. Syst., Man, Cybern., B Cybern.*, vol. 36, no. 3, pp. 559–570, Jun. 2006.

[31] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Comput. Electron. Agricult.*, vol. 147, pp. 70–90, Apr. 2018.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[33] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," 2015, *arXiv:1508.04025*.

[34] Q. Wu, H. Zhang, X. Gao, P. He, P. Weng, H. Gao, and G. Chen, "Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems," in *Proc. World Wide Web Conf.*, May 2019, pp. 2091–2102.

[35] S. Zhai, K.-H. Chang, R. Zhang, and Z. M. Zhang, "DeepIntent: Learning attentions for online advertising with recurrent neural networks," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1295–1304.

[36] J. Chen, B. Sun, H. Li, H. Lu, and X.-S. Hua, "Deep CTR prediction in display advertising," in *Proc. 24th ACM Int. Conf. Multimedia*, 2016, pp. 811–820.

[37] R. C. Moore, "On log-likelihood-ratios and the significance of rare events," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2004, pp. 333–340.

[38] R. He and J. McAuley, "Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering," in *Proc. 25th Int. Conf. World Wide Web*, 2016, pp. 507–517.

[39] L. Yan, W.-J. Li, G.-R. Xue, and D. Han, "Coupled group lasso for web-scale CTR prediction in display advertising," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 802–810.

[40] J. McAuley, C. Targett, Q. Shi, and A. Van Den Hengel, "Image-based recommendations on styles and substitutes," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2015, pp. 43–52.

[41] G. Zhou, N. Mou, Y. Fan, Q. Pi, W. Bian, C. Zhou, X. Zhu, and K. Gai, "Deep interest evolution network for click-through rate prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 1, 2019, pp. 5941–5948.

[42] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.

[43] F. Li, Z. Chen, P. Wang, Y. Ren, D. Zhang, and X. Zhu, "Graph intention network for click-through rate prediction in sponsored search," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2019, pp. 961–964.

[44] Q. Pi, W. Bian, G. Zhou, X. Zhu, and K. Gai, "Practice on long sequential user behavior modeling for click-through rate prediction," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 2671–2679.

[45] Z. Xiao, L. Yang, W. Jiang, Y. Wei, Y. Hu, and H. Wang, "Deep multi-interest network for click-through rate prediction," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 2265–2268.

[46] W. Jiang, Y. Jiao, Q. Wang, C. Liang, L. Guo, Y. Zhang, Z. Sun, Y. Xiong, and Y. Zhu, "Triangle graph interest network for click-through rate prediction," in *Proc. 15th ACM Int. Conf. Web Search Data Mining*, Feb. 2022, pp. 401–409.

JIANGUO WEI is currently a Lecturer with the School of Computer Engineering, Weifang University, Weifang, China. His current research interests include multimedia content analysis and retrieval, machine learning, and data mining.

LEI WANG received the Ph.D. degree from the School of Computer Science and Technology, Shandong University, in 2010. He is currently an Associate Professor with the School of Computer Engineering, Weifang University, China. His research interests include intelligent manufacturing systems and machine learning.

MEILING GE received the master's degree from the College of Computer Science, Shaanxi Normal University. She is currently a Teaching Assistant with the School of Computer Engineering, Weifang University, Weifang, China. Her current research interests include recommendation, multimedia content analysis and retrieval, machine learning, and data mining.